BioNLP 2024

The 23rd Meeting
# of the ACL Special Interest Group
on Biomedical Natural Language Processing

Proceedings of the Workshop and Shared Tasks

August 16, 2024

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
209 N. Eighth Street
Stroudsburg, PA 18360
USA
Tel: +1-570-476-8006
Fax: +1-570-476-0860
acl@aclweb.org

ISBN 979-8-89176-130-8

Biomedical natural language processing in 2024: The year of BioMedGen

Dina Demner-Fushman, Sophia Ananiadou, Makoto Miwa, Kirk Roberts and Jun-ichi Tsujii

The development of Large Language Models (LLMs) applied to complex Biomedical Language Processing tasks keeps growing steadily. This growth instigated the anticipation of major breakthroughs in language generation and downstream healthcare tasks, as well as concerns with respect to potential harms and irresponsible use of AI applications. Both the medical informatics communities and regulatory agencies are developing guidelines and checklists for conducting trustworthy LLM-based research and reporting the results of this research [1] [2] [3].

The submissions to the BioNLP 2024 workshop and the Shared Tasks demonstrated once again that the workshop sponsored by the ACL Special Interest Group on Biomedical Natural Language Processing (SIGBIOMED) is the preferred venue for the groundbreaking research and applications in Biomedical Language Processing. BioNLP remains the flagship and the generalist in biomedical language processing, accepting all noteworthy work independently of the tasks and languages studied. The quality of submissions continues to impress the program committee and the organizers.

BioNLP 2024 received 61 submissions, of which six were accepted for oral presentation and 37 as poster presentations. The presentations cover a wide range of the foundational biomedical language processing research and clinical applications, exploring generation of a variety of clinical reports, extraction of information from the literature and social media, prediction of patients' outcomes and generation of datasets and benchmarks for question answering.

The Shared Tasks included generation of radiology reports (RRG24, 8 participating teams), generation of hospital course summaries and discharge instructions (Discharge Me!, 12 participating teams), and abstractive summarization of biomedical articles (BioLaySumm, 14 participating teams). The overviews of the tasks and short presentations of the best performing approaches are included in the workshop program. The participants in all Shared Tasks present their work in a dedicated poster session.

The keynote by Titipat Achakulvisut, Department of Biomedical Engineering, Mahidol University, Thailand is titled Enhancing Neuroscience Conferences through Natural Language Processing

This talk presents the development and implementation of natural language processing (NLP) tools at neuroscience conferences. Dr. Achakulvisut has successfully integrated these tools into various conferences, including a recommendation engine at the Society for Neuroscience (SfN) meeting, one-on-one matching at the Conference on Cognitive Computational Neuroscience (CCN), paper-reviewer matching for the Computational and Systems Neuroscience (COSYNE) conference, and reviewer recommendations for NBDT journal. His group employs a fine-tuning and contrastive learning approach to adapt transformer-based models, such as MiREAD and SciBERT for neuroscience. The models were evaluated using both distance metrics and recommendation arena assessments. The goal of exploring NLP tools in non-computer science domains is to enhance the interactions of researchers and attendees.

As always, we are deeply grateful to the authors of the submitted papers and to the reviewers (listed elsewhere in this volume) who produced three thorough and thoughtful reviews for each paper in a fairly short review period. The quality of submitted work continues to grow, and the organizers are truly grateful to the members of our amazing Program Committee, who helped us to determine which work was ready to be presented, and which would benefit from the additional experiments and analyses suggested by the reviewers. As in years past, we are looking forward to a productive workshop and

[1] https://www.coalitionforhealthai.org/

[2] https://www.fda.gov/media/153486/download

[3] https://tripod-llm.vercel.app/

hoping it will foster new collaborations and research. This will enable our community to continue making valuable contributions to public health and well-being and clinical research.

The advent of Generative AI and LLMs has also transformed our workshop introducing new challenges and opportunities. We are now in the era of BioMedGen.

Organizing Committee

Organizers:

Dina Demner-Fushman, US National Library of Medicine
Sophia Ananiadou, National Centre for Text Mining and University of Manchester, UK
Makoto Miwa, Toyota Technological Institute, Japan
Kirk Roberts, UTHealth, Houston, Texas, USA
Junichi Tsujii, National Institute of Advanced Industrial Science and Technology, Japan

Program Committee:

Joseph Damilola Akinyemi, University of Ibadan, Ibadan, Nigeria
Sophia Ananiadou, National Centre for Text Mining and University of Manchester, UK
Emilia Apostolova, Anthem, Inc.
Eiji Aramaki, University of Tokyo, Japan
Berry de Bruijn, National Research Council Canada
Leonardo Campillos Llanos, Universidad Autonoma de Madrid, Spain
Jiyu Chen, CSIRO Data61, Australia
Mike Conway, University of Utah, USA
Surabhi Datta, Melax Tech, USA Dina Demner-Fushman, US National Library of Medicine
Dmitriy Dligach, Loyola University Chicago, USA
Kathleen C. Fraser, National Research Council Canada
Yanjun Gao, University of Wisconsin-Madison, USA
Natalia Grabar, CNRS, France
Cyril Grouin, LIMSI - CNRS, France
Tudor Groza, The Garvan Institute of Medical Research, Australia
Deepak Gupta, US National Library of Medicine
Thierry Hamon, LIMSI-CNRS, France
Sam Henry, Christopher Newport University, USA
William Hogan, UCSD, USA
Brian Hur, University of Washington, USA
Richard Jackson, AstraZeneca
Antonio Jimeno Yepes, IBM, Melbourne Area, Australia
Sarvnaz Karimi, CSIRO, Australia
Nazmul Kazi, Montana State University, USA
Won Gyu Kim, US National Library of Medicine
Roman Klinger, University of Stuttgart, Germany
Anna Koroleva, Omdena Andre Lamurias, University of Lisbon, Portugal
Majid Latifi, University of York, York, UK
Alberto Lavelli, FBK-ICT, Italy
Robert Leaman, US National Library of Medicine
Lung-Hao Lee, National Central University, Taiwan
Ulf Leser, Humboldt Universitat zu Berlin, Germany
Diwakar Mahajan, MIT-IBM Watson AI Lab, USA
Timothy Miller, Boston Children's Hospital and Harvard Medical School, USA
Makoto Miwa, Toyota Technological Institute, Japan
Claire Nedellec, INRA, France
Guenter Neumann, DFKI, Saarland, Germany
Aurelie Neveol, LIMSI - CNRS, France
Mariana Neves, German Federal Institute for Risk Assessment, Germany
Brian Ondov, Yale University, USA

Laura Plaza, Universidad Nacional de Educacion a Distancia, Spain
Noon Pokaratsiri Goldstein, DFKI, Germany
Francois Remy, Ghent University, Belgium
Francisco J. Ribadas-Pena, University of Vigo, Spain
Roland Roller, DFKI GmbH, Berlin, Germany
Mourad Sarrouti, Sumitovant Biopharma, Inc., USA
Peng Su, University of Delaware, USA
Madhumita Sushil, University of California, San Francisco, USA
Mario Sanger, Humboldt Universitat zu Berlin, Germany
Andrew Taylor, Yale University School of Medicine, USA
Karin Verspoor, RMIT University, Australia
Davy Weissenbacher, Cedars-Sinai, Los Angeles, California, USA
Nathan M. White, James Cook University, Australia
W John Wilbur, US National Library of Medicine
Amelie Wuhrl, University of Stuttgart, Germany
Dongfang Xu, Cedars-Sinai, USA.
Shweta Yadav, University of Illinois Chicago, USA
Jingqing Zhang, Imperial College London, UK
Ayah Zirikly, Johns Hopkins, USA.
Pierre Zweigenbaum, LIMSI - CNRS, France

Table of Contents

Conference Program

Friday, August 16, 2024

08:15–08:30 Opening remarks

08:30–10:30 Session 1: Oral Presentations

08:30–08:50 *Improving Self-training with Prototypical Learning for Source-Free Domain Adaptation on Clinical Text*
Seiji Shimizu, Shuntaro Yada, Lisa Raithel and Eiji ARAMAKI

08:50–09:10 *Generation and Evaluation of Synthetic Endoscopy Free-Text Reports with Differential Privacy*
Agathe Zecevic, Xinyue Zhang, Sebastian Zeki and Angus Roberts

09:10–09:30 *Evaluating the Robustness of Adverse Drug Event Classification Models using Templates*
Dorothea MacPhail, David Harbecke, Lisa Raithel and Sebastian Möller

09:30–09:50 *Advancing Healthcare Automation: Multi-Agent System for Medical Necessity Justification*
Himanshu Gautam Pandey, Akhil Amod and Shivang Kumar

09:50–10:10 *Open (Clinical) LLMs are Sensitive to Instruction Phrasings*
Alberto Mario Ceballos-Arroyo, Monica Munnangi, Jiuding Sun, Karen Zhang, Jered McInerney, Byron C. Wallace and Silvio Amir

10:10–10:30 *Analysing zero-shot temporal relation extraction on clinical notes using temporal consistency*
Vasiliki Kougia, Anastasiia Sedova, Andreas Joseph Stephan, Klim Zaporojets and Benjamin Roth

10:30–11:00 ***Coffee Break***

Friday, August 16, 2024 (continued)

11:00–13:00 Session 2: Shared Tasks

11:00–11:20 *Overview of the First Shared Task on Clinical Text Generation: RRG24 and "Discharge Me!"*
Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz and Jean-Benoit Delbrouck

11:20–11:25 *e-Health CSIRO at RRG24: Entropy-Augmented Self-Critical Sequence Training for Radiology Report Generation*
Aaron Nicolson, Jinghui Liu, Jason Dowling, Anthony Nguyen and Bevan Koopman

11:25–11:30 *WisPerMed at "Discharge Me!”: Advancing Text Generation in Healthcare with Large Language Models, Dynamic Expert Selection, and Priming Techniques on MIMIC-IV*
Hendrik Damm, Tabea Margareta Grace Pakull, Bahadır Eryılmaz, Helmut Becker, Ahmad Idrissi-Yaghir, Henning Schäfer, Sergej Schultenkämper and Christoph M. Friedrich

11:30–11:50 *Overview of the BioLaySumm 2024 Shared Task on the Lay Summarization of Biomedical Research Articles*
Tomas Goldsack, Carolina Scarton, Matthew Shardlow and Chenghua Lin

11:50–12:00 *UIUC_BioNLP at BioLaySumm: An Extract-then-Summarize Approach Augmented with Wikipedia Knowledge for Biomedical Lay Summarization*
Zhiwen You, Shruthan Radhakrishna, Shufan Ming and Halil Kilicoglu

12:00–12:20 *Shared Tasks Discussion*

12:20–13:00 *Invited Talk – Titipat Achakulvisut: Enhancing Neuroscience Conferences through Natural Language Processing*

13:00–14:30 *Lunch*

Friday, August 16, 2024 (continued)

14:00–15:30 BioNLP Poster Session

End-to-End Relation Extraction of Pharmacokinetic Estimates from the Scientific Literature
Ferran Gonzalez Hernandez, Victoria Smith, Quang Nguyen, Palang Chotsiri, Thanaporn Wattanakul, José Antonio Cordero, Maria Rosa Ballester, Albert Sole, Gill Mundin, Watjana Lilaonitkul, Joseph F. Standing and Frank Kloprogge

KG-Rank: Enhancing Large Language Models for Medical QA with Knowledge Graphs and Ranking Techniques
Rui Yang, Haoran Liu, Edison Marrese-Taylor, Qingcheng Zeng, Yuhe Ke, Wanxin Li, Lechao Cheng, Qingyu Chen, James Caverlee, Yutaka Matsuo and Irene Li

MedExQA: Medical Question Answering Benchmark with Multiple Explanations
Yunsoo Kim, Jinge Wu, Yusuf Abdulle and Honghan Wu

Do Clinicians Know How to Prompt? The Need for Automatic Prompt Optimization Help in Clinical Note Generation
Zonghai Yao, Ahmed Jaafar, Beining Wang, Zhichao Yang and hong yu

Domain-specific or Uncertainty-aware models: Does it really make a difference for biomedical text classification?
Aman Sinha, Timothee Mickus, Marianne Clausel, Mathieu Constant and Xavier Coubez

Can Rule-Based Insights Enhance LLMs for Radiology Report Classification? Introducing the RadPrompt Methodology.
Panagiotis Fytas, Anna Breger, Ian Selby, Simon Baker, Shahab Shahipasand and Anna Korhonen

Using Large Language Models to Evaluate Biomedical Query-Focused Summarisation
Hashem Hijazi, Diego Molla, Vincent Nguyen and Sarvnaz Karimi

Continuous Predictive Modeling of Clinical Notes and ICD Codes in Patient Health Records
Mireia Hernandez Caralt, Clarence Boon Liang Ng and Marek Rei

Can GPT Redefine Medical Understanding? Evaluating GPT on Biomedical Machine Reading Comprehension
Shubham Vatsal and Ayush Singh

Get the Best out of 1B LLMs: Insights from Information Extraction on Clinical Documents
Saeed Farzi, Soumitra Ghosh, Alberto Lavelli and Bernardo Magnini

K-QA: A Real-World Medical Q&A Benchmark
Itay Manes, Naama Ronn, David Cohen, Ran Ilan Ber, Zehavi Horowitz-Kugler and Gabriel Stanovsky

Friday, August 16, 2024 (continued)

Large Language Models for Biomedical Knowledge Graph Construction: Information extraction from EMR notes
Vahan Arsenyan, Spartak Bughdaryan, Fadi Shaya, Kent Wilson Small and Davit Shahnazaryan

Document-level Clinical Entity and Relation extraction via Knowledge Base-Guided Generation
Kriti Bhattarai, Inez Y. Oh, Zachary B. Abrams and Albert M. Lai

BiCAL: Bi-directional Contrastive Active Learning for Clinical Report Generation
Tianyi Wu, Jingqing Zhang, Wenjia Bai and Kai Sun

Generation and De-Identification of Indian Clinical Discharge Summaries using LLMs
Sanjeet Singh, Shreya Gupta, Niralee Gupta, Naimish Sharma, Lokesh Srivastava, Vibhu Agarwal and Ashutosh Modi

Pre-training data selection for biomedical domain adaptation using journal impact metrics
Mathieu LAI-KING and Patrick Paroubek

Leveraging LLMs and Web-based Visualizations for Profiling Bacterial Host Organisms and Genetic Toolboxes
Gilchan Park, Vivek Mutalik, Christopher Neely, Carlos Soto, Shinjae Yoo and Paramvir Dehal

REAL: A Retrieval-Augmented Entity Linking Approach for Biomedical Concept Recognition
Darya Shlyk, Tudor Groza, Marco Mesiti, Stefano Montanelli and Emanuele Cavalleri

Is That the Right Dose? Investigating Generative Language Model Performance on Veterinary Prescription Text Analysis
Brian Hur, Lucy Lu Wang, Laura Hardefeldt and Meliha Yetisgen

MiDRED: An Annotated Corpus for Microbiome Knowledge Base Construction
William Hogan, Andrew Bartko, Jingbo Shang and Chun-Nan Hsu

Do Numbers Matter? Types and Prevalence of Numbers in Clinical Texts
Rahmad Mahendra, Damiano Spina, Lawrence Cavedon and Karin Verspoor

A Fine-grained citation graph for biomedical academic papers: the finding-citation graph
Yuan Liang, Massimo Poesio and Roonak Rezvani

Evaluating Large Language Models for Predicting Protein Behavior under Radiation Exposure and Disease Conditions
Ryan Engel and Gilchan Park

Friday, August 16, 2024 (continued)

XrayGPT: Chest Radiographs Summarization using Large Medical Vision-Language Models
Omkar Chakradhar Thawakar, Abdelrahman M. Shaker, Sahal Shaji Mullappilly, Hisham Cholakkal, Rao Muhammad Anwer, Salman Khan, Jorma Laaksonen and Fahad Khan

Multilevel Analysis of Biomedical Domain Adaptation of Llama 2: What Matters the Most? A Case Study
Vicente Ivan Sanchez Carmona, Shanshan Jiang, Takeshi Suzuki and Bin Dong

Mention-Agnostic Information Extraction for Ontological Annotation of Biomedical Articles
Oumaima El Khettari, Noriki Nishida, Shanshan Liu, Rumana Ferdous Munne, Yuki Yamagata, Solen Quiniou, Samuel Chaffron and Yuji Matsumoto

Automatic Extraction of Disease Risk Factors from Medical Publications
Maxim Rubchinsky, Ella Rabinovich, Adi Shribman, Netanel Golan, Tali Sahar and Dorit Shweiki

Intervention extraction in preclinical animal studies of Alzheimer's Disease: Enhancing regex performance with language model-based filtering
YIYUAN PU, Kaitlyn Hair, Daniel Beck, Mike Conway, Malcolm MacLeod and Karin Verspoor

Efficient Biomedical Entity Linking: Clinical Text Standardization with Low-Resource Techniques
Akshit Achara, Sanand Sasidharan and Gagan N

XAI for Better Exploitation of Text in Medical Decision Support
Ajay Madhavan Ravichandran, Julianna Grune, Nils Feldhus, Aljoscha Burchardt, Roland Roller and Sebastian Möller

Optimizing Multimodal Large Language Models for Detection of Alcohol Advertisements via Adaptive Prompting
Daniel Cabrera Lozoya, Jiahe Liu, Simon D'Alfonso and Mike Conway

Extracting Epilepsy Patient Data with Llama 2
Ben Holgate, Shichao Fang, Anthony Shek, Matthew McWilliam, Pedro Viana, Joel S. Winston, James T. Teo and Mark P. Richardson

How do you know that? Teaching Generative Language Models to Reference Answers to Biomedical Questions
Bojana Bašaragin, Adela Ljajić, Darija Medvecki, Lorenzo Cassano, Miloš Košprdić and Nikola Milošević

Low Resource ICD Coding of Hospital Discharge Summaries
Ashton Williamson, David de Hilster, Amnon Meyers, Nina Hubig and Amy Apon

Towards ML-supported Triage Prediction in Real-World Emergency Room Scenarios
Faraz Maschhur, Klaus Netter, Sven Schmeier, Katrin Ostermann, Rimantas Palunis, Tobias Strapatsas and Roland Roller

Creating Ontology-annotated Corpora from Wikipedia for Medical Named-entity Recognition
Johann Frei and Frank Kramer

Paragraph Retrieval for Enhanced Question Answering in Clinical Documents
Vojtech Lanz and Pavel Pecina

15:30–16:00 ***Coffee Break***

16:00–17:50 **Shared Tasks Poster Session**

RRG24

CID at RRG24: Attempting in a Conditionally Initiated Decoding of Radiology Report Generation with Clinical Entities
Yuxiang Liao, Yuanbang Liang, Yipeng Qin, Hantao Liu and Irena Spasic

MAIRA at RRG24: A specialised large multimodal model for radiology report generation
Shaury Srivastav, Mercy Ranjit, Fernando Pérez-García, Kenza Bouzid, Shruthi Bannur, Daniel C. Castro, Anton Schwaighofer, Harshita Sharma, Maximilian Ilse, Valentina Salvatelli, Sam Bond-Taylor, Fabian Falck, Anja Thieme, Hannah Richardson, Matthew P. Lungren, Stephanie L. Hyland and Javier Alvarez-Valle

AIRI at RRG24: LLaVa with specialised encoder and decoder
Marina Munkhoeva, Dmitry Umerenkov and Valentin Samokhin

iHealth-Chile-1 at RRG24: In-context Learning and Finetuning of a Large Multimodal Model for Radiology Report Generation
Diego Campanini, Oscar Loch, Pablo Messina, Rafael Elberg and Denis Parra

iHealth-Chile-3&2 at RRG24: Template Based Report Generation
Oscar Loch, Pablo Messina, Rafael Elberg, Diego Campanini, Álvaro Soto, René Vidal and Denis Parra

Gla-AI4BioMed at RRG24: Visual Instruction-tuned Adaptation for Radiology Report Generation
Xi Zhang, Zaiqiao Meng, Jake Lever and Edmond S.L. Ho

SICAR at RRG2024: GPU Poor's Guide to Radiology Report Generation
Kiartnarin Udomlapsakul, Parinthapat Pengpun, Tossaporn Saengja, Kanyakorn Veerakanjana, Krittamate Tiankanon, Pitikorn Khlaisamniang, Pasit Supholkhan, Amrest Chinkamol, Pubordee Aussavavirojekul, Hirunkul Phimsiri, tara sripo, Chiraphat Boonnag, Trongtum Tongdee, Thanongchai Siriapisith, Pairash Saiviroonporn, Jiramet Kinchagawat and Piyalitt Ittichaiwong

Friday, August 16, 2024 (continued)

BIoLaySum

Friday, August 16, 2024 (continued)

HGP-NLP at BioLaySumm: Leveraging LoRA for Lay Summarization of Biomedical Research Articles using Seq2Seq Transformers
Hemang Malik, Gaurav Pradeep and Pratinav Seth

Ctyun AI at BioLaySumm: Enhancing Lay Summaries of Biomedical Articles Through Large Language Models and Data Augmentation
siyu bao, ruijing zhao, Siqin Zhang, jinghui zhang, weiyin wang and yunian ru

17:50–18:00 Closing remarks

Improving Self-training with Prototypical Learning for Source-Free Domain Adaptation on Clinical Text

Seiji Shimizu[1] **Shuntaro Yada**[1] **Lisa Raithel**[2,3,4] **Eiji Aramaki**[1]
[1]Nara Institute of Science and Technology (NAIST)
[2]BIFOLD – Berlin Institute for the Foundations of Learning and Data
[3]Quality & Usability Lab, Technische Universität Berlin
[4]German Research Center for Artificial Intelligence (DFKI)
shimizu.seiji.so8@is.naist.jp

Abstract

Domain adaptation is crucial in the clinical domain since the performance of a model trained on one domain (source) degrades seriously when applied to another domain (target). However, conventional domain adaptation methods often cannot be applied due to data sharing restrictions on source data. Source-Free Domain Adaptation (SFDA) addresses this issue by only utilizing a source model and unlabeled target data to adapt to the target domain. In SFDA, self-training is the most widely applied method involving retraining models with target data using predictions from the source model as pseudo-labels. Nevertheless, this approach is prone to contain substantial numbers of errors in pseudo-labeling and might limit model performance in the target domain. In this paper, we propose a Source-Free Prototype-based Self-training (SFPS) aiming to improve the performance of self-training. SFPS generates prototypes without accessing source data and utilizes them for prototypical learning, namely prototype-based pseudo-labeling and contrastive learning. Also, we compare entropy-based, centroid-based, and class-weights-based prototype generation methods to identify the most effective formulation of the proposed method. Experimental results across various datasets demonstrate the effectiveness of the proposed method, consistently outperforming vanilla self-training. The comparison of various prototype-generation methods identifies the most reliable generation method that improves the source model persistently. Additionally, our analysis illustrates SFPS can successfully alleviate errors in pseudo-labeling.

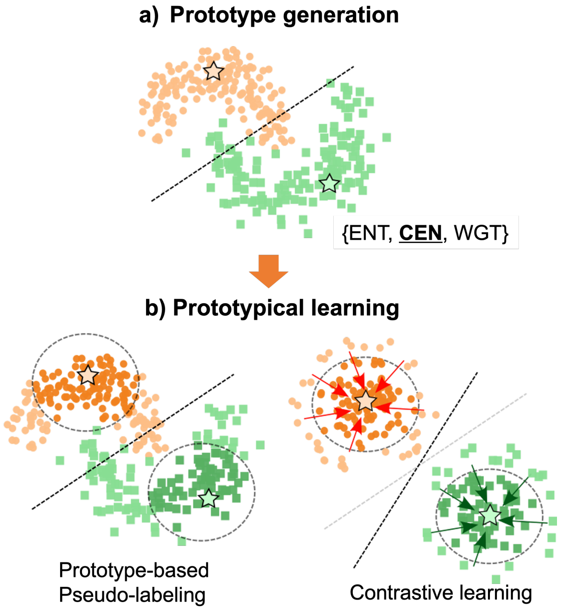


Figure 1: Illustration of SFPS. ⋆ denotes prototypes, and dotted lines denote the model's decision boundaries. First, we generate prototypes with either the entropy-based (**ENT**), centroid-based (**CEN**), or class-weights-based (**WGT**) method (a). **CEN** is chosen in this example. Then, we utilize these prototypes for prototypical learning (b), consisting of prototype-based pseudo-labeling and contrastive learning to update the source model and obtain distinct representations of target data.

1 Introduction

Domain adaptation is crucial in Clinical Natural Language Processing (Clinical NLP) since it is known that the performance of the model trained on one domain (source) degrades seriously on another domain (target) in the face of domain shifts such as different specialty or institution's formatting (Wu et al., 2014; Bethard et al., 2017; Miller et al., 2017). Despite the significant advancements in research on domain adaptation, most existing methods assume access to the labeled source data (Kouw and Loog, 2019; Ramponi and Plank, 2020). This assumption is frequently violated in the clinical domain, where data sharing is restricted due to patients' privacy concerns (Laparra et al., 2020). Source-Free Domain Adaptation (SFDA) addresses this issue by only utilizing a source model and unlabeled target

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 1–13
August 16, 2024. ©2024 Association for Computational Linguistics

data to adapt to the target domain (Liang et al., 2020; Chidlovskii et al., 2016).

Self-training (Kumar et al., 2010; Li and Zhang, 2019) has been shown to be a versatile and effective method for SFDA in computer vision (Yu et al., 2023). In Clinical NLP, a shared task was newly introduced in SemEval 2021 Task 10 (Laparra et al., 2021), prompting the development of SFDA methods on clinical text. Active learning and self-training combined with data augmentation emerged as widely applied methods (Su et al., 2021). The systematic comparison of the proposed methods indicates that active learning can reliably improve the source model's performance, while self-training is unreliable, failing to consistently outperform the source model (Su et al., 2022). However, active learning requires additional annotation on the target data, which can be difficult due to the expertise required for the annotation (Su et al., 2021). This necessitates improvement of the existing self-training method that does not rely on either additional annotation or source data for adapting the source model in Clinical NLP.

Prototypical learning (Snell et al., 2017; Wang et al., 2022) can potentially improve self-training, yet existing methods are not applicable in the SFDA setting. In general, self-training involves retraining the source model with target data by assigning the predictions from the source model as pseudo-labels. Nevertheless, pseudo-labels assigned in this manner contain a substantial number of errors and might limit the model's performance. Prototypical learning, such as prototype-based pseudo-labeling (Gu, 2020), and contrastive learning (Li et al., 2021) are proven to be effective for improving self-training (Yang et al., 2023; Mou et al., 2023; Zhou et al., 2023). However, existing methods assume access to labeled source data to generate reliable prototypes, making them inapplicable in source-free settings. How to generate reliable prototypes without accessing source data remains unanswered.

In this paper, we aim to provide answers to the following questions:

Q1: *Can prototypical learning improve self-training in **SFDA**?*

Q2: *Which method can generate reliable prototypes in the absence of labeled source data?*

Q3: *Is prototypical learning effective for alleviating errors in pseudo-labeling?*

To answer **Q1**, we introduce source-free prototype-based self-training (SFPS). Unlike existing methods, we generate prototypes without accessing source data (Fig. 1a) and leverage the generated prototypes for prototypical learning (Fig. 1b) consisting of prototype-based pseudo-labeling (Gu, 2020) and contrastive learning (Li et al., 2021) to alleviate errors in pseudo-labeling. To answer **Q2**, we explore three source-free prototype generation methods, namely entropy-based, centroid-based, and class-weights-based methods, inspired by the works in computer vision (Kim et al., 2021; Liang et al., 2020; Ding et al., 2024). To answer **Q3**, we compare the pseudo-label quality of SFPS and vanilla self-training.

We conduct experiments on negation detection and time expression recognition tasks from SemEval2021 Task10 with the source models trained on clinical texts. Our experimental results show the effectiveness of SFPS, outperforming vanilla self-training methods in most datasets. A comparison of various prototype-generation methods reveals that the centroid-based generation method can reliably improve the source model performance among other generation methods. We evaluate the pseudo-label quality of the proposed method and demonstrate the proposed method could successfully alleviate the errors in pseudo-labeling.

To summarize, we provide answers to the above questions as follows:

A1: *Prototypical learning can improve self-training in SFDA and consistently outperform vanilla self-training.*

A2: *Centroid-based prototype generation can reliably improve model performance without accessing the source data.*

A3: *Prototypical learning effectively alleviates the errors in pseudo-labels.*

2 Related Work

2.1 Source-Free Domain Adaptation

Source-free domain adaptation (SFDA) only uses a source model and unlabeled target data to adapt the model to the target domain. In recent years, SFDA has gained significant traction in computer vision. Various methods have been proposed, such as virtual domain generation (Tian et al., 2022), image style translation (Luan et al., 2017), and neighborhood clustering (Yang et al., 2021). Among them, self-training (Kumar et al., 2010; Li and Zhang,

2019) has been proven to be versatile and effective (Yu et al., 2023).

In contrast, SFDA methods in NLP are comparatively limited. Yin et al. (2022) introduced the SFDA method in question answering. They utilized an additional masking module during source model training and froze some weights of the masking module during self-training to maintain domain invariant knowledge. Zhang et al. (2021) aligned joint distributions between a trained source model and target domain samples using joint maximum mean discrepancy during knowledge distillation. In Clinical NLP, SemEval-2021 Task10 (Laparra et al., 2021) introduced a shared task for SFDA consisting of negation detection and time expression recognition. Only source model and unlabeled target data were provided to the participants. Self-training, active learning, and data augmentation methods were proposed. Although active learning can reliably improve the source model (Su et al., 2022), this method requires additional annotation on target data, which can be difficult due to data sharing restriction and expertise required for the annotation (Su et al., 2021). Hence, we extend the self-training method by developing the SFDA method, which is feasible in a wider range of situations.

2.2 Prototypical Learning

Prototypical learning, which aims to summarize a class by representative prototypes, has been widely used in semi-supervised and unsupervised learning (Wang et al., 2022; Snell et al., 2017). In self-training, pseudo-labeling based on the source model predictions suffers from errors. To improve self-training, prototype-based pseudo labeling (Gu, 2020) combined with contrastive learning (Li et al., 2021) are employed in semi-supervised learning (SSL) and unsupervised domain adaptation. Prototype-based pseudo-labeling assigns labels based on the similarities/distances between prototypes and target data representations instead of relying on model prediction. Contrastive learning enhances representations of target data by facilitating the formation of clusters of prototypes and text representations.

Yang et al. (2023) applied prototype-based pseudo-labeling and contrastive learning for text classification in SSL setting. They defined a centroid of the labeled data as a class-specific prototype and assigned pseudo-labels to unlabeled samples based on their distances from prototypes. These prototypes were then utilized as anchors to create high-density clusters of text representations via contrastive learning. In zero-shot cross-lingual named entity recognition, Zhou et al. (2023) defined the moving average of labeled data as a class prototype and used them for pseudo-labeling and contrastive learning. Mou et al. (2023) introduced prototype-based pseudo-labeling and contrastive learning in out-of-distribution intent classification. They defined randomly initialized embeddings as prototypes and updated them using embedded text representations of samples belonging to the same class. While prototype-based pseudo-labeling and contrastive learning have shown effectiveness in sentence and token classification, existing methods assume access to source data, making them inapplicable in the SFDA setting.

2.3 Source-free Prototype Generation

In the field of computer vision, various source-free prototype-generation methods have been proposed. Kim et al. (2021) defined samples with low entropy as prototypes and leveraged them for unsupervised learning by assigning pseudo-labels based on the distance between target image representations and prototypes. Liang et al. (2020) obtained the centroid of each class based on source model outputs. Pseudo-labels are assigned to unlabeled target data based on the distance between the class centroid and target samples. They further employ information maximization between target image representation and classifier output to update the model encoder. Ding et al. (2024) used the weights of the source classifier as class prototypes, constructing a class-balanced proxy source domain. The proxy source domain is then used for an inter-domain mixup that aligns the proxy domain and the target domain. While these works have independently combined source-free prototype generation with various SFDA techniques, we systematically compare the effectiveness of different prototype generation methods. Specifically, we combine various prototype generation methods with prototype-based pseudo-labeling and contrastive learning.

3 Problem Definition

Unlike conventional domain adaptation, we only have access to the source model and unlabeled target data in SFDA. Let $c \in C$ be a class from the set of all classes of interest, M the source model, and $X = \{x_0, ..., x_n\}$ the target data where n is the number of target samples. In general, source-free self-training aims to improve the performance of M using pseudo-labels $\hat{y}_i \in C$ assigned to x_i. How $\hat{y}_i$ is assigned and leveraged for the improvement depends on an individual method. However, only M and X are available for the adaptation.

For the generalizability of our study, the only assumption we make on the source model M is that it can be decomposed into an encoder (denoted by F) and classifier (denoted by G), i.e., $M := G(F(\cdot))$.

Strictly speaking, the definition of x_i differs between sentence classification and token classification. For sentence classification, one sample is equivalent to one input, i.e., $x_i = seq_i$ where seq is a sentence. For token classification, one sample contains a series of inputs, i.e., $x_i = [w_0^i, ..., w_m^i]$ where w is a token and m is a sentence length. For convenience, we use x_i to denote both a sentence and a token input.

4 Methodology

This section presents the proposed source-free prototype-based self-training (SFPS). The conceptual workflow is shown in Figure 2. First, we generate class prototypes from unlabeled target data with a source model (composed of F and G) using prototype generation (Section 4.1). Then, we utilize generated prototypes for prototypical learning, consisting of prototype-based pseudo-labeling and contrastive learning. Prototype-based pseudo-labeling assigns pseudo-labels based on the similarity between prototypes and text representations (Section 4.2). Contrastive learning improves the representation of target data by increasing/decreasing similarity between prototypes and target representations belonging to the same/a different class(Section 4.3). We describe the overall algorithm (Section 4.4) with the variants of SFPS.

4.1 Prototype generation

We generate a set of prototypes for a class $c \in C$ using only M and X. We experiment with three different source-free prototype generation methods, namely entropy-based (**ENT**), centroid-based (**CEN**), and class-weights-based (**WGT**) methods. In each method, we construct a set of prototypes for c, which is denoted by $\Phi_c = \{\phi_0^c, ..., \phi_K^c\}$ where K is the number of prototypes.

ENT: The entropy-based method chooses the representations of samples with high entropy as prototypes. Following Kim et al. (2021), we first calculate the lowest entropy for each class and set the largest value among them as a threshold (denoted by η), which is calculated by:

$$\begin{aligned}&\eta = \max\{\min(\mathcal{H}_c)|c \in C\},\\ &\mathcal{H}_c = \{H(x_i)|x_i \in X_c\}\end{aligned} \tag{1}$$

where $H(x_i)$ denotes the entropy of x_i given by M, and X_c denotes the set of samples predicted as c by M.

Then, a set of prototypes is generated by:

$$\Phi_c = \{F(x_i)|x_i \in X, H(x_i) \leq \eta\} \tag{2}$$

CEN: The centroid-based method chooses the centroid for each class as a prototype generated by:

$$\Phi_c = \frac{1}{|X_c|} \sum_{x_i \in X_c} F(x_i) \tag{3}$$

WGT: The class-weights-based method chooses the weights of G corresponding to c as a prototype. Inspired by Ding et al. (2024), we also include the top $K-1$ most similar text representations $F(x_i)$ (denoted by $\mathcal{X}_c$) with the class-specific weights as prototypes. A set of prototypes is generated by:

$$\begin{aligned}&\Phi_c = \mathcal{X}_c \cup \{w_c\}\\ &\mathcal{X}_c = \{F(x_i)|x_i \in \max_{x_i;K-1}(\text{sim}(F(x_i), w_c))\}\end{aligned} \tag{4}$$

where w_c denotes the corresponding weights and $\max_{x_i;K-1}(\text{sim}(\cdot))$ denotes choosing top $K-1$ samples with maximum similarity for each class. In this work, we use cosine similarity as a similarity measure (denoted by sim).

4.2 Prototype-based Pseudo-labeling

For prototype-based pseudo labeling, we find the most similar Φ_c to x_i and assign c as a label. Since relying on a single prototype can be unstable due to the unsupervised nature of the prototype generation method, we assign the label based on the prototype set Φ_c. To do so, we first calculate the similarity score s_c between Φ_c and x_i:

$$s_c(x_i) = \frac{1}{|\Phi_c|} \sum_k \text{sim}(\phi_k^c, F(x_i)) \tag{5}$$

and assign a pseudo-label by:

$$\hat{y}_i = \operatorname*{argmax}_c s_c(x_i), \forall c \in C \tag{6}$$

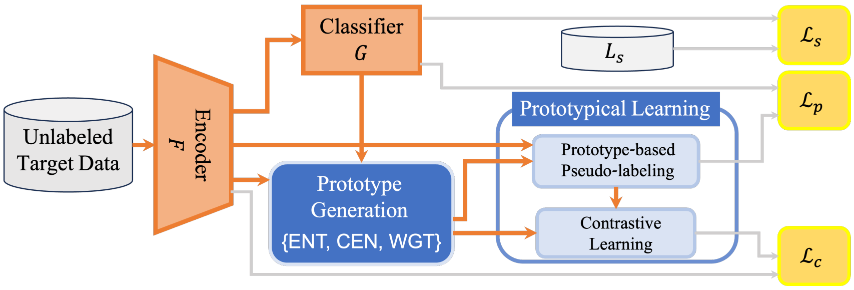


Figure 2: The workflow of SFPS. Prototypes are generated based on unlabeled target data and the source model (Section 4.1) and utilized for prototypical learning, consisting of prototype-based pseudo-labeling (Section 4.2) and contrastive learning (Section 4.3). This flow is illustrated by orange arrows. Three learning objectives are used for fine-tuning. $\mathcal{L}_p$ is an unsupervised loss between the pseudo-labels and the model predictions (Eq. 7). $\mathcal{L}_c$ is a contrastive loss based on the distance between prototypes and text representations (Eq. 9). $\mathcal{L}_s$ is a regularization loss based on an un-updated source model predictions (denoted by L_s) and the model predictions (Eq. 11).

Based on $\hat{y}_i$, the learning objective for fine-tuning M is given by:

$$\mathcal{L}_p = -\frac{1}{n}\sum_{i}^{n} \mathbb{1}(\hat{y}_i = c) \log \frac{\exp(p_i^c)}{\sum_{c \in C} \exp(p_i^c)} \quad (7)$$

where $\mathbb{1}(\cdot)$ is a indicator function and p_i^c is the predicted probability for the class c given by M with respect to x_i.

4.3 Contrastive Learning

Contrastive learning aims to obtain a distinct representation of x_i by increasing the similarity between $F(x_i)$ and prototypes of the same class while decreasing the similarity for the prototypes of the different classes. Inspired by Zhou et al. (2023), we employ the moving average of Φ_c per batch to update the representation of x_i, which is calculated by:

$$\mu_c = \alpha \frac{1}{|\Phi_c|} \sum_{k} \phi_k^c + (1-\alpha) \frac{1}{|B|} \sum_{i} F(x_i), \quad \forall i \in \{i | \hat{y}_i = c\}, \quad (8)$$

where α denotes the hyperparameter controlling the degree of updates and $|B|$ denotes the number of inputs per batch. In this way, we can ensure further stability of updates since μ_c is dynamically changing in accordance with $F(x_i)$ throughout the fine-tuning. Based on μ_c, we update $F(x_i)$ by the contrastive learning objective given by:

$$\mathcal{L}_c {=} - \sum_{i,c} \log \mathbb{1}(\hat{y}_i {=} c) \frac{\exp(\mathrm{sim}(F(x_i), \mu_c)/\beta)}{\sum_C \exp(\mathrm{sim}(F(x_i), \mu_c)/\beta)} \quad (9)$$

where β is a temperature coefficient.

4.4 Overall Algorithm

Algorithm 1 describes the whole process of SFPS. In line 14, we construct a set of pseudo-labels based on confidence scores. For sentence classification, we use the similarity score (Eq. 5) as a confidence score, i.e., confidence $= s_c(x_i)$. For token classification, x_i is a single token in a sentence. We take the average similarity scores of tokens for each sentence and use it as a confidence score. The confidence score for token classification is given by:

$$\text{confidence} = \frac{1}{m}\sum_{i}^{m} s_c(x_i) \quad (10)$$

Following Kim et al. (2021), we use pseudo-labels $\hat{y}_i^0$ given by the un-updated source model $M_0(x_i)$ as a regularizer so that the model does not diverge too much from the original source model (in line 15). The regularizer learning objective is given by:

$$\mathcal{L}_s = -\frac{1}{n}\sum_{i}^{n} \mathbb{1}(\hat{y}_i^0 = c) \log \frac{\exp(p_i^c)}{\sum_{c \in C} \exp(p_i^c)} \quad (11)$$

The overall objective $\mathcal{L}$ is the sum of Eq. 7, 9 and 11, namely:

$$\mathcal{L} = \mathcal{L}_p + \mathcal{L}_s + \mathcal{L}_c \quad (12)$$

Su et al. (2022) compared various formulations of self-training by changing the maximum number of iterations, the data construction strategy, and the model training strategy as parameters. Following this, we change the parameters of Algorithm 1 below to investigate which combination of prototype generation, data construction strategy, and model training strategy is most effective.

T the maximum number of iterations.

S_D the data construction strategy: KD to keep the training data from the previous iteration, or RD to reset.

S_M the model training strategy: KM to keep the model from the previous iteration, or RM to reset.

S_G the prototype generation methods: **ENT**, **CEN** or **WGT** to use entropy-based, centroid-based or class-weights-based method.

Algorithm 1: SFPS

```
Input :
  $M$: the source-domain model
  $X$: the target domain data
  $T$: the maximum number of iterations
  $L_p$: the pseudo-labels assigned via Eq. 6
  $L_s$: the pseudo-labels assigned by the un-updated source model
  $S_D$: the data construction strategy
  $S_M$: the model training strategy
  $S_P$: the prototype generation strategy
$M_0 \leftarrow$ Copy($M$)
$X_0 \leftarrow$ Copy($X$)
$L_p \leftarrow \emptyset$
for $t \leftarrow 0$ to $T$ do
    if $X = \emptyset$ then
        Stop training
    end
    if $S_D = RD$ then
        $L_p = \emptyset$
        $X = X_0$
    end
    Get $\Phi_c$ by Eq. 2, 3 or 4 based on $S_P$
    Get $s_c$ and $\hat{y}$ by Eq. 5 and 6
    $L_p \leftarrow \{(x_i, \hat{y}_i)$ for $x_i \in X$ if confidence $> \tau\}$
    $L_s \leftarrow \{(x_i, \hat{y}_i^0)$ for $(x_i, \hat{y}_i) \in L_p\}$
    if $L_{p_t} = \emptyset$ or $L_{p_t} = L_{p_{t-1}}$ then
        Stop training
    end
    if $S_D = KD$ then
        $X \leftarrow X - \{x_i$ for $(x_i, \hat{y}_i) \in L_{p_t}\}$
    end
    if $S_M = RM$ then
        $M \leftarrow M_0$
    end
    Fine-tune $M$ given $\Phi_c$, $L_p$, and $L_s$, using Eq. 12
end
```

5 Experiments

We conduct experiments with negation detection and time expression recognition datasets and compare a fully fine-tuned model (Oracle), an unadapted source model (Source), all vanilla self-training variants in Su et al. (2022) (Vanilla), and variants of SFPS. Vanilla and SFPS **do not utilize labeled target data** because our target problem setting is SFDA. However, datasets used in the experiments are fully annotated and used to train Oracle models.

We note that we do not expect SFPS to outperform Oracle. We consider Oracle as a upper bound for the performance in each dataset.

5.1 Datasets

We use the target data and source models from SemEval2021 Task10: negation detection and time expression recognition (Laparra et al., 2021). The provided source models were fine-tuned using English RoBERTa-base (Liu et al., 2019) as base models.

As described in Su et al. (2022), these two tasks are suitable for SFDA because (1) source data is difficult to share, (2) target data can not be easily annotated due to the complexity of the annotation task, and (3) models suffer a large performance loss in the face of domain shift in these tasks.

The negation detection task involves the classification of an event within a context span (indicated by special tokens "*<e>*" and "*</e>*") as in below.

She did not complain of <e> any fever </e>

This task aims to correctly predict whether "*any fever*" is negated or not. The source model for this task was trained using Mayo Clinic clinical notes. Two target data for this task are clinical notes from Partners HealthCare's participation in the i2b2 2010 Challenge (**i2b2**) and ICU progress notes from Beth Israel in the MIMIC-III corpus (**MIMIC**).

The time expression recognition task involves sequence tagging, aiming to identify time entities in a document and assign them SCATE types (Bethard and Parker, 2016). An example sentence is given below.

the patient underwent surgery for gallstones on July 14, 2019

The goal of this task is to predict "*July*" as *Month-Of-Year*, "*14*" as *Day-Of-Month* and "*2019*" as *Year*. The source model for this task was trained using clinical notes from Mayo Clinic as a part of SemEval 2018 Task 6 (Laparra et al., 2018). Two target datasets for this task are news articles from SemEval 2018 Task 6 (**News**) and reports from food security warning systems, including the UN World Food Programme and the Famine Early Warning Systems Network (**Food**).

We used the same development-test split as in Su et al. (2022) for all datasets as shown in Table 1. Note that the unit of numbers is a sentence for negation detection and a document for time expression recognition. Each document is preprocessed into sentences in time expression recognition.

	MIMIC	**i2b2**	**News**	**Food**
Dev	1916	1109	20	4
Test	7664	4436	79	13

Table 1: The number of development and test data. The unit is a sentence for negation detection (**MIMIC** and **i2b2**) and a document for time expression recognition (**News** and **Food**). Development sets are used for the adaptation.

5.2 Implementation Details

We used PyTorch[1] for the implementation of SFPS. For the preprocessing and implementation of vanilla self-training methods, we used the provided scripts from Su et al. (2022)[2]. We set the hyperparameters for SFPS, K, α, β, and τ to be 10, 0.9 and 0.5 respectively. We set the maximum number of iterations T to be 1 or 30. For a fair comparison, all the hyperparameters for fine-tuning except for learning rate are the same as in source model training and used for both SFPS and vanilla self-training. Since the proposed method has more learning objectives, we set the learning rate to 1.0×10^{-5} for SFPS and kept the original learning rate of 5.0×10^{-5} for vanilla self-training models. Other hyperparameters used for both SFPS and vanilla self-training are summarized in the Appendix A.1.

5.3 Results

We evaluated all models using the same evaluation metrics (F1, precision, and recall) as in Su et al. (2022). The results are the average of five different seeds. Due to limited space, we only present F1 scores (in percentage points) in Table 2. We provide the full results in Appendix A.2.

Strategy	**MIMIC**	**i2b2**	**News**	**Food**
Oracle	88.9	92.3	85.1	87.6
Source	63.5	84.6	79.1	78.5
Vanilla				
Single	67.4	87.1	79.1	77.4
KD+KM	66.5	87.6	79.3	77.7
KD+RM	68.7	87.6	79.2	78.2
RD+KM	55.4	87.8⋆	79.0	77.9
RD+RM	67.9	87.3	79.2	77.8
SFPS$_{\mathrm{ENT}}$				
Single	**71.3**	85.5	79.3	**78.9**
KD+KM	68.4	86.3	77.1	78.2
KD+RM	66.1	86.0	77.2	78.2
RD+KM	66.6	86.6	**80.1**⋆	**78.7**
RD+RM	53.6	86.7	**79.9**	**78.9**
SFPS$_{\mathrm{CEN}}$				
Single	**70.3**	84.8	**79.4**	**79.2**⋆
KD+KM	66.8	85.1	76.8	**79.2**⋆
KD+RM	67.8	85.8	79.0	**78.8**
RD+KM	63.6	86.8	**79.9**	77.2
RD+RM	67.6	87.5	**79.8**	**78.5**
SFPS$_{\mathrm{WGT}}$				
Single	**71.8**⋆	85.2	78.5	**78.3**
KD+KM	66.6	85.9	78.5	78.1
KD+RM	66.6	86.0	78.5	74.9
RD+KM	65.7	86.8	78.5	78.2
RD+RM	64.7	86.3	78.5	75.7

Table 2: The results of the experiment in F1 scores. Oracle, Source, and Vanilla denote the fully fine-tuned model, the un-updated source model, and vanilla self-training variants, respectively. KD, RD, KM, and RM are the variations due to the choice of the training strategies (see Section 4.4). $T = 1$ for Single and $T = 30$ for the others. The strategies that outperformed the source model are underlined. The scores above the best-performing vanilla method are in bold. Scores with a star are the best among all the self-training methods.

Several formulations of SFPS are shown to be effective. The best-performing SFPS formulations outperformed the best-performing vanilla self-training methods in most datasets (3 out of 4). In **MIMIC**, the best-performing formulation for SFPS was **WGT** with Single with 3.1 points higher F1 score than the best vanilla self-training method. In **i2b2**, no prototype-based method outperformed the best-performing vanilla self-training method. **CEN** with RD+RM has the highest F1 score among other SFPS formulations and scored 0.3 points below the best-performing vanilla self-training method. Given that the F1 score achieved by the best-performing vanilla self-training method is already high and relatively close to the Oracle, achieving further improvement may be challenging without the availability of labeled target data.

[1] https://pytorch.org/

[2] https://github.com/xinsu626/SourceFreeDomainAdaptation

In **News**, the best-performing SFPS formulation was **ENT** with RD+KM improving 0.8 points in F1 score from the best vanilla method. In **Food**, the best combination was **CEN** with Single and KD+KM, with an F1 score 1.0 point higher than the best vanilla method.

6 Discussion

Experimental results indicate that **CEN** with Single can reliably improve the source model compared with vanilla self-training. While no vanilla self-training method could outperform the un-updated source models in all datasets, **CEN** with Single outperformed the un-updated source model in all datasets and the best-performing vanilla self-training model in 3 out of 4 datasets. Since labeled target data is not available (or difficult to obtain) in SFDA, hyperparameter tuning is not realistic. Hence, it is important for an SFDA method to consistently outperform the source model regardless of task and dataset.

In the following section, we show that SFPS can properly alleviate the errors in pseudo-labeling (Section 6.1). We also conducted an ablation study to show both contrastive learning and regularization are effective for improving model performance (Section 6.2).

6.1 Pseudo-label Quality

Although we did not use any labeled data for adaptation, labels of the target data are available for all the datasets. In order to compare the pseudo-label qualities of the un-updated source model, vanilla self-training, and SFPS, we calculate the accuracy and macro F1 score of the pseudo-labeling by best-performing models of each method. The results are shown in Table 3. SFPS have the highest accuracy in all datasets and F1 score in 3 out of 4 datasets, indicating that prototype-based pseudo-labeling combined with contrastive learning could successfully alleviate the errors in pseudo-labeling.

6.2 Ablation study

In order to investigate the effectiveness of the contrastive learning objective ($\mathcal{L}_c$) and the regularization term ($\mathcal{L}_s$) in Eq.12, we conduct an ablation study. We compared the performance of the model with (1) all objectives (Full), (2) without contrastive learning ($-\mathcal{L}_c$), (3) without regularization ($-\mathcal{L}_s$), and (4) only unsupervised learning objective ($-\mathcal{L}_c - \mathcal{L}_s$). Table 4 shows the results on four datasets in F1 score. In most datasets, the contrastive learning objective or regularization term alone improves the performance compared with only using unsupervised learning with pseudo labels. With the exception of **MIMIC**, Full models have the highest F1 scores, indicating that the contrastive learning objective combined with the regularization term is effective for improving model performance.

	MIMIC		**i2b2**	
	ACC	**F1**	**ACC**	**F1**
Source	93.5	77.2	93.2	88.9
Vanilla	93.7	77.9	**94.0**	90.2
SFPS	**94.5**	**81.9**	**94.0**	**90.4**
	News		**Food**	
	ACC	**F1**	**ACC**	**F1**
Source	98.4	50.7	**95.9**	56.2
Vanilla	98.3	50.0	95.8	**56.4**
SFPS	**98.5**	**52.1**	**95.9**	56.3

Table 3: Pseudo-labeling accuracy and F1 score on development data. **Source** and **Vanilla** denotes the un-updated source model and the best-performing vanilla self-training model. SFPS successfully alleviates the errors in pseudo-labeling compared with vanilla self-training.

Objectives	**MIMIC**	**i2b2**	**News**	**Food**
Full	71.8	**87.5**	**80.1**	**79.2**
$-\mathcal{L}_c$	**72.3**	86.1	79.4	78.2
$-\mathcal{L}_s$	70.8	77.8	78.4	77.7
$-\mathcal{L}_c - \mathcal{L}_s$	71.5	46.8	78.4	77.7

Table 4: Results of ablation study in F1 score. Using both the contrastive-learning objective and the regularization term is effective in most of the datasets.

7 Conclusion

In this paper, we proposed source-free prototype-based self-training (SFPS) composed of prototype generation, prototype-based pseudo labeling, and contrastive learning. We compared entropy-based, centroid-based, and class-weights-based methods to identify the most reliable prototype generation method. We conducted experiments with two negation detection datasets and two time expression recognition datasets. Experimental results show the effectiveness of SFPS, consistently outperforming vanilla self-training. The comparison of various prototype generation methods reveals that the centroid-based generation method combined with a single iteration strategy is the most reliable formulation, outperforming the source model in all datasets and the best vanilla self-training model in 3 out of 4 datasets. Also, our analysis demonstrates that the proposed method can successfully alleviate errors in pseudo-labeling.

8 Limitations

We show that the proposed SFPS has an advantage over vanilla self-training methods in negation detection and time expression recognition tasks. However, this work has several limitations:(1) Experiments are only conducted on clinical/English corpora, limiting the generalizability of the results to other domains.; (2) The conclusions stated in this paper are based only on empirical evidence. Hence, they lack a theoretical analysis; (3) The gap between fully fine-tuned models and the proposed method is still large. This is expected, considering that the proposed method does not utilize labeled data at all for the adaptation. Yet, the model performance can be improved via task-specific modules such as class balancing for time expression recognition; (4) Although we tackled both sentence and token classification, the tasks employed in this experiment are limited in number. It is desirable to test the effectiveness of the proposed method in other tasks in the clinical domain and other domains as well.

Acknowledgements

This work was supported by JST CREST Grant Number JPMJCR22N1, and Cross-ministerial Strategic Innovation Promotion Program (SIP) on "Integrated Health Care System" Grant Number JPJ012425, Japan. We are grateful for the support of the German Research Foundation (DFG) under the trilateral ANR-DFG-JST AI Research project KEEPHA (DFG-442445488), and the German Federal Ministry of Education and Research under the grant BIFOLD24B.

References

Steven Bethard and Jonathan Parker. 2016. A semantically compositional annotation scheme for time normalization. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3779–3786, Portorož, Slovenia. European Language Resources Association (ELRA).

Steven Bethard, Guergana Savova, Martha Palmer, and James Pustejovsky. 2017. SemEval-2017 task 12: Clinical TempEval. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 565–572, Vancouver, Canada. Association for Computational Linguistics.

Boris Chidlovskii, Stephane Clinchant, and Gabriela Csurka. 2016. Domain adaptation in the absence of source domain data. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 451–460, New York, NY, USA. Association for Computing Machinery.

Yuhe Ding, Lijun Sheng, Jian Liang, Aihua Zheng, and Ran He. 2024. Proxymix: Proxy-based mixup training with label refinery for source-free domain adaptation. *Neural Netw.*, 167(C):92–103.

Xiaowei Gu. 2020. A self-training hierarchical prototype-based approach for semi-supervised classification. *Information Sciences*, 535:204–224.

Youngeun Kim, Donghyeon Cho, Kyeongtak Han, Priyadarshini Panda, and Sungeun Hong. 2021. Domain adaptation without source data. *IEEE Transactions on Artificial Intelligence*, 2(6):508–518.

Wouter M. Kouw and Marco Loog. 2019. An introduction to domain adaptation and transfer learning.

Abhishek Kumar, Avishek Saha, and Hal Daume. 2010. Co-regularization based semi-supervised domain adaptation. In *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc.

Egoitz Laparra, Steven Bethard, and Timothy A Miller. 2020. Rethinking domain adaptation for machine learning over clinical language. *JAMIA Open*, 3(2):146–150.

Egoitz Laparra, Xin Su, Yiyun Zhao, Özlem Uzuner, Timothy Miller, and Steven Bethard. 2021. SemEval-2021 task 10: Source-free domain adaptation for semantic processing. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 348–356, Online. Association for Computational Linguistics.

Egoitz Laparra, Dongfang Xu, Ahmed Elsayed, Steven Bethard, and Martha Palmer. 2018. SemEval 2018 task 6: Parsing time normalizations. In *Proceedings of the 12th International Workshop on Semantic Evaluation*, pages 88–96, New Orleans, Louisiana. Association for Computational Linguistics.

Junnan Li, Pan Zhou, Caiming Xiong, and Steven Hoi. 2021. Prototypical contrastive learning of unsupervised representations. In *International Conference on Learning Representations*.

Limin Li and Zhenyue Zhang. 2019. Semi-supervised domain adaptation by covariance matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(11):2724–2739.

Jian Liang, Dapeng Hu, and Jiashi Feng. 2020. Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6028–6039. PMLR.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Fujun Luan, Sylvain Paris, Eli Shechtman, and Kavita Bala. 2017. Deep photo style transfer.

Timothy Miller, Dmitriy Dligach, Steven Bethard, Chen Lin, and Guergana Savova. 2017. Towards generalizable entity-centric clinical coreference resolution. *Journal of Biomedical Informatics*, 69:251–258.

Yutao Mou, Xiaoshuai Song, Keqing He, Chen Zeng, Pei Wang, Jingang Wang, Yunsen Xian, and Weiran Xu. 2023. Decoupling pseudo label disambiguation and representation learning for generalized intent discovery. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9661–9675, Toronto, Canada. Association for Computational Linguistics.

Alan Ramponi and Barbara Plank. 2020. Neural unsupervised domain adaptation in NLP—A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6838–6855, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4080–4090, Red Hook, NY, USA. Curran Associates Inc.

Xin Su, Yiyun Zhao, and Steven Bethard. 2021. The University of Arizona at SemEval-2021 task 10: Applying self-training, active learning and data augmentation to source-free domain adaptation. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 458–466, Online. Association for Computational Linguistics.

Xin Su, Yiyun Zhao, and Steven Bethard. 2022. A comparison of strategies for source-free domain adaptation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8352–8367, Dublin, Ireland. Association for Computational Linguistics.

Jiayi Tian, Jing Zhang, Wen Li, and Dong Xu. 2022. Vdm-da: Virtual domain modeling for source data-free domain adaptation. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(6):3749–3760.

Haobo Wang, Ruixuan Xiao, Yixuan Li, Lei Feng, Gang Niu, Gang Chen, and Junbo Zhao. 2022. PiCO: Contrastive label disambiguation for partial label learning. In *International Conference on Learning Representations*.

Stephen Wu, Timothy Miller, James Masanz, Matt Coarr, Scott Halgrim, David Carrell, and Cheryl Clark. 2014. Negation's not solved: Generalizability versus optimizability in clinical natural language processing. *PLOS ONE*, 9(11):1–11.

Shiqi Yang, Yaxing Wang, Joost van de Weijer, Luis Herranz, and Shangling Jui. 2021. Generalized source-free domain adaptation.

Weiyi Yang, Richong Zhang, Junfan Chen, Lihong Wang, and Jaein Kim. 2023. Prototype-guided pseudo labeling for semi-supervised text classification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16369–16382, Toronto, Canada. Association for Computational Linguistics.

M. Yin, B. Wang, Y. Dong, and C. Ling. 2022. Source-free domain adaptation for question answering with masked self-training.

Zhiqi Yu, Jingjing Li, Zhekai Du, Lei Zhu, and Heng Tao Shen. 2023. A comprehensive survey on source-free domain adaptation.

Bo Zhang, Xiaoming Zhang, Yun Liu, Lei Cheng, and Zhoujun Li. 2021. Matching distributions between model and data: Cross-domain knowledge distillation for unsupervised domain adaptation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5423–5433, Online. Association for Computational Linguistics.

Ran Zhou, Xin Li, Lidong Bing, Erik Cambria, and Chunyan Miao. 2023. Improving self-training for cross-lingual named entity recognition with contrastive and prototype learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4018–4031, Toronto, Canada. Association for Computational Linguistics.

A Appendix

A.1 Hyperparameters

Except for the learning rate, we used the same hyperparameters for both vanilla self-training and prototype-based self-training. For both negation detection and time expression recognition tasks, we set the hyperparameters to the same values as Su et al. (2022), which are summarized in Table 5 and 6.

Hyperparameter	Value
maximum sequence length	128
batch size	8
epochs	10
gradient accumulation steps	4
learning rate warm up steps	0
weight decay	0.0
adam epsilon	1.0×10^{-8}
maximum gradient norm	1.0

Table 5: Hyperparameters for negation detection

Hyperparameter	Value
maximum sequence length	271
batch size	2
epochs	3
gradient accumulation steps	1
learning rate warm up steps	500
weight decay	0.01
adam epsilon	1.0×10^{-8}
maximum gradient norm	1.0

Table 6: Hyperparameters for time expression recognition

All models were trained on AdamW (Loshchilov and Hutter, 2019) and a single NVIDIA Quadro RTX 8000 GPU. A training process took about 30 minutes per fine-tuning.

A.2 Full Results

The full results (in percentage points) for negation detection are presented in Table 7, and the results for time expression recognition are presented in Table 8.

Strategy	MIMIC			i2b2		
	F1	P	R	F1	P	R
Oracle	88.9	88.4	89.5	92.3	93.3	91.3
Source	63.5	93.8	48.0	84.6	92.6	77.9
Vanilla						
Single	67.4	94.7	52.4	87.1	95.1	80.4
KD+KM	66.5	95.4	51.1	87.6	94.4	81.8
KD+RM	68.7	95.4	53.6	87.6	95.3	81.0
RD+KM	55.4	75.7	43.7	87.8*	94.2	82.3
RD+RM	67.9	95.5	52.6	87.3	94.6	81.2
SFPS$_{\mathbf{ENT}}$						
Single	**71.3**	90.4	58.9	85.5	88.9	82.4
KD+KM	68.4	92.7	54.3	86.3	94.1	79.7
KD+RM	66.1	95.0	50.8	86.0	93.0	79.9
RD+KM	66.6	94.5	51.5	86.6	92.9	81.1
RD+RM	53.6	74.8	41.8	86.7	92.1	81.9
SFPS$_{\mathbf{CEN}}$						
Single	**70.3**	89.0	58.3	84.8	89.5	80.7
KD+KM	66.8	95.1	51.6	85.1	91.6	79.6
KD+RM	67.8	92.9	53.5	85.8	93.2	79.5
RD+KM	63.6	96.2	47.7	86.8	93.4	81.1
RD+RM	67.6	92.7	53.5	87.5	94.1	81.8
SFPS$_{\mathbf{WGT}}$						
Single	**71.8***	88.7	60.3	85.2	88.6	82.0
KD+KM	66.6	94.5	51.7	85.9	93.0	79.9
KD+RM	66.6	93.8	51.8	86.0	93.9	79.4
RD+KM	65.7	95.0	50.3	86.8	93.8	80.8
RD+RM	64.7	93.4	50.4	86.3	90.7	82.3

Table 7: The results in the negation detection task. Oracle, Source, and Vanilla denote the fully fine-tuned model, an unadapted source model, and vanilla self-training variants, respectively. $T = 1$ for Single and $T = 30$ for the others. The strategies outperformed the source model are underlined. The scores above the best-performing vanilla method are in bold. Scores with a star are the best among all the self-training methods.

Strategy	**News**			**Food**		
	F1	P	R	F1	P	R
Oracle	85.1	85.4	84.8	87.6	85.7	89.7
Source	79.1	79.5	78.7	78.5	82.9	74.6
Vanilla						
Single	79.1	79.8	78.5	77.4	80.7	74.5
KD+KM	79.3	78.4	80.2	77.7	79.9	75.7
KD+RM	79.2	78.4	80.0	78.2	80.8	75.7
RD+KM	79.0	78.1	79.9	77.9	80.0	75.8
RD+RM	79.2	78.3	80.1	77.8	80.0	75.8
SFPS$_{\textbf{ENT}}$						
Single	79.3	80.9	77.7	**78.9**	84.8	73.9
KD+KM	77.1	82.4	72.5	78.2	87.5	70.8
KD+RM	77.2	81.7	73.2	78.2	87.3	70.9
RD+KM	**80.1***	80.8	79.3	**78.7**	83.2	74.7
RD+RM	**79.9**	81.2	78.7	**78.9**	83.7	74.6
SFPS$_{\textbf{CEN}}$						
Single	**79.4**	81.2	77.6	**79.2***	85.0	74.1
KD+KM	76.8	81.4	72.6	**79.2***	86.9	72.9
KD+RM	79.0	81.3	76.9	**78.8**	85.7	73.0
RD+KM	**79.9**	79.8	80.0	77.2	78.9	75.7
RD+RM	**79.8**	80.5	79.1	**78.5**	82.4	75.0
SFPS$_{\textbf{WGT}}$						
Single	78.5	80.8	76.3	**78.3**	84.2	73.2
KD+KM	78.5	80.8	76.3	78.1	84.4	72.7
KD+RM	78.5	80.8	76.3	74.9	88.3	65.2
RD+KM	78.5	80.8	76.3	78.2	84.1	73.0
RD+RM	78.5	80.8	76.3	75.7	88.3	66.6

Table 8: The results in time expression recognition task. Oracle, Source, and Vanilla denote the fully fine-tuned model, an un-adapted source model, and vanilla self-training variants, respectively $T = 1$ for Single and $T = 30$ for the others. The strategies that outperformed the source model are underlined. The scores above the best-performing vanilla method are in bold. Scores with a star are the best among all the self-training methods.

Generation and Evaluation of Synthetic Endoscopy Free-Text Reports with Differential Privacy

Agathe Zecevic*[1,2], Xinyue Zhang*[3], Sebastian Zeki[1], Angus Roberts[3]
[1]Gastroenterology Department, Guy's and St Thomas' NHS Foundation Trust, United Kingdom
[2]Clinical Scientific Computing, Guy's and St Thomas' NHS Foundation Trust, United Kingdom
[3]Biostatistics and Health Informatics, Institute of Psychiatry, Psychology and Neuroscience
King's College London, United Kingdom
agathe.zecevic@gstt.nhs.uk, leo.xinyue.zhang@kcl.ac.uk, **Joint first authorship*

Abstract

The development of NLP models in the healthcare sector faces important challenges due to the limited availability of patient data, mainly driven by privacy concerns. This study proposes the generation of synthetic free-text medical reports, specifically focusing on the gastroenterology domain, to address the scarcity of specialised datasets, while preserving patient privacy. We fine-tune BioGPT on over 90 000 endoscopy reports and integrate Differential Privacy (DP) into the training process. 10 000 DP-private synthetic reports are generated by this model. The generated synthetic data is evaluated across multiple dimensions: similarity to real datasets, language quality, and utility in both supervised and semi-supervised NLP tasks. Results suggest that while DP integration impacts text quality, it offers a promising balance between data utility and privacy, improving the performance of a real-world downstream task. Our study underscores the potential of synthetic data to facilitate model development in the healthcare domain without compromising patient privacy.

1 Introduction

The development of computer-aided tools in medicine, including natural language processing (NLP), requires real patient data for model training. However, this development has been significantly limited due to the lack of availability of patient data due to privacy concerns, restricted access to hospital data, a scarcity of labeled data, barriers to sharing pretrained models, and a lack of capable computational resources in many healthcare settings (Wu et al., 2022). The lack of specialised datasets when developing NLP models can lead to biased or ungeneralizable models (Panch et al., 2019; Daneshjou et al., 2021). Recent literature highlights that open-source, synthetic datasets could mitigate data scarcity and lead to robust AI model training, particularly in NLP (Ive et al., 2020). However, very few studies tackle the generation of synthetic free text in the medical domain, with no known studies focusing on gastroenterology text reports.

While synthetic data presents a viable solution to dataset scarcity, ensuring the privacy of patient data in the original dataset used for training remains essential. Recent findings suggest that simply de-identifying the training set by removing names and unique identifiers is insufficient to prevent patient re-identification (Sarkar et al., 2024). Despite this, it is not common practice to include a robust data privacy framework when generating synthetic medical data (Begoli et al., 2018; Guan et al., 2021). To maintain stringent patient confidentiality, our approach incorporates Differential Privacy (DP), a framework that mathematically guarantees the level of inability to identify an individual's data within a dataset (Dwork et al., 2006). Our approach is motivated by the fact that only a limited number of academic papers investigate the application of differential privacy in the generation of synthetic data within healthcare (Klymenko et al., 2022).

The quality and utility of these generated reports must also be rigorously assessed to ensure their practical application in clinical settings. Utility, in our context, refers to the degree to which the synthetic data can be used to perform real-world tasks, such as text classification. It is crucial to compare how differential privacy impacts the quality and utility of synthetic data and whether it can be used to enhance performance on various tasks. These tasks can be supervised, such as text classification, unsupervised or semi-supervised, like Task Adaptive Pre-Training Tasks (TAPT).

2 Aims

- We create free text endoscopy reports generated with differential privacy by fine-tuning a medical domain GPT-based model.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 14–24
August 16, 2024. ©2024 Association for Computational Linguistics

- We assess the similarity of DP-generated reports to the original patient data (training dataset), using a set of experiments that includes outliers re-generation.

- We aim to quantify the potential quality reduction induced by DP by assessing the text quality of synthetic text reports with and without DP.

- We assess and compare the utility of DP-generated synthetic reports across supervised and semi-supervised tasks.

3 Related Work

Generating Synthetic Medical Notes with Differential Privacy. The generation of synthetic text data with Differential Privacy (DP) is an emerging field with limited research. While (Yue et al., 2022) has provided a comprehensive framework for generating synthetic text data with DP, none of the investigated datasets include medical data. Sarkar et al. (2024) also propose a framework for synthetic data generation with DP.

Privacy Assessment of Synthetic Medical Notes and Text Similarity. Numerous studies have shown that Large Language Models (LLMs) and Generative Models can efficiently produce synthetic text reports (Melamud and Shivade, 2019; Abdollahi et al., 2021; Li et al., 2021; Guan et al., 2018; Tang et al., 2023). However, the privacy aspect of the synthetic data is often overlooked or relies on simple downstream analyses. A common practice in studies involving synthetic patient text data, especially in those not using DP, is to employ metrics like the Hamming distance or the Levenshtein distance to assess the privacy level of generated reports. These methods measure how closely synthetic data can be linked to their original counterparts. A threshold is established, and synthetic reports are considered vulnerable if their distances fall below this threshold (Ghosheh et al., 2024; Yan et al., 2021; Zhang et al., 2020).

Text Quality Assessment of Synthetic Medical Notes. The text quality of synthetic reports is often evaluated on a per-report basis, using metrics (Zhou et al., 2023) such as BLEU, ROUGE, or BERTScore (Zhang et al., 2019). However, these measurements require a set of references with which to compare the synthetic text for report-level evaluation. For synthetic text that does not have references, research tends to measure the distribution similarity on a corpus level using metrics such as generation perplexity (Fan et al., 2018), self-BLEU (Zhu et al., 2018) or Mauve (Pillutla et al., 2021). However, these methods do not give a score for each single report. In our recent work (in process of publication), we trained a language quality model that scores any generated report without the need for a reference text. The model is trained on a dataset that is corrupted by shuffling and inflection of real text. The model learns the mapping from each corrupted text, which can be seen as a proxy for model output, to a quality score, which is calculated by comparing the corrupted text with its original, unaltered form. This approach has proven to align well with human judgment and is effective in distinguishing higher-quality real texts from synthetic counterparts of lower language quality based on the generated scores.

Synthetic Data Utility. Sarkar et al. (2024) assess the utility of DP-generated synthetic reports through downstream tasks. However, their downstream tasks focus on ICD-10 code classification models trained on synthetic data, which differs significantly from our study. We explore the utility of DP-generated synthetic data both in a supervised setting, using it for data augmentation, and in a semi-supervised setting, employing it for further pre-training of the classifiers, which has not previously been documented in the literature.

4 Methods

A summary of the overall pipeline is depicted in Figure 1.

4.1 Data Access

Inclusion criteria Endoscopy reports were extracted from the electronic patient records (EPR) of St Thomas' Hospital in London. Data acquisition was authorised through an institutional board review. The dataset includes the following unfiltered procedures: Colonoscopy, Gastroscopy, Endoscopic ultrasound (EUS), Sigmodoiscopy and Endoscopic retrograde cholangiopancreatography (ERCP). The records spanned from January 2017 to October 2023.

Exclusion criteria To ensure patient privacy and comply with UK health service national data

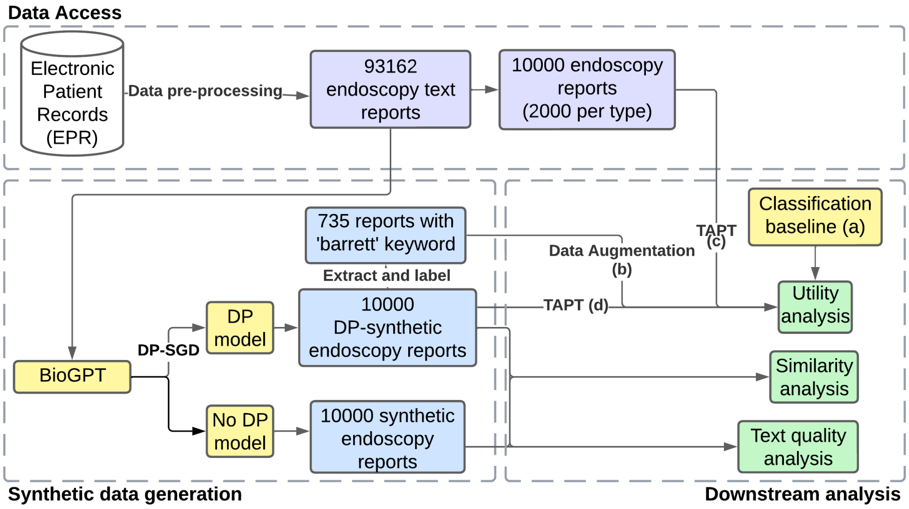

Data Access
Electronic Patient Records (EPR)
Data pre-processing
93162 endoscopy text reports
10000 endoscopy reports (2000 per type)
735 reports with 'barrett' keyword
Extract and label
Data Augmentation (b)
TAPT (c)
Classification baseline (a)
Utility analysis
DP-SGD
DP model
10000 DP-synthetic endoscopy reports
TAPT (d)
BioGPT
No DP model
10000 synthetic endoscopy reports
Similarity analysis
Text quality analysis
Synthetic data generation
Downstream analysis

Figure 1: Summary of the methodology of the study. (a), (b), (c), (d) correspond to the experiments described in Section 4.8.

opt-out policy (nhs, 2023), all individuals who had explicitly opted out of having their data used for research purposes were excluded from the study.

A total of **93162** reports were included, representing a diverse range of gastrointestinal conditions and providing a comprehensive dataset for the generation of synthetic endoscopy text reports.

4.2 Data Pre-processing and De-identification

Our dataset was anonymized, as required by NHS England and the UK Information Commissioner's Office (ICO)'s anonymisation code of practice[1]. Direct identifiers (such as names, addresses, and contact numbers) and indirect identifiers (such as clinician names and dates) were systematically removed from the dataset without replacement using regular expressions.

The remaining pre-processing was performed using the EndoMineR package[2], a tool designed for the analysis of free-text in endoscopy reports (Zeki, 2018). The package enabled the extraction of relevant sections from endoscopy reports.

4.3 Differential Privacy

Differential Privacy ensures that the output of a randomized function applied to a dataset is statistically indistinguishable, up to a specified degree of error, regardless of whether any single individual's data is included in the dataset or not. The notion of (ϵ, δ)-differential privacy, as defined by (Dwork et al., 2006) and further elaborated in recent literature (Yue et al., 2022) is as follows:
A randomized function F provides (ϵ, δ)-differential privacy if for all datasets D_1 and D_2 differing on at most one element, and for all subsets S of the possible outputs of F:

$$\Pr[F(D_1) \in S] \leq e^{\epsilon} \times \Pr[F(D_2) \in S] + \delta \quad (1)$$

ϵ (epsilon), also called *privacy budget*, is a small non-negative parameter that quantifies the strength of the privacy guarantee. δ (delta), typically close to zero, represents the small probability that the ϵ-differential privacy guarantee may be exceeded. ϵ is a key feature of differential privacy: a lower value guarantees greater privacy but generally reduces the utility of the generated data. In this study, we set $\epsilon = 4$, and δ to

$$\delta = \frac{1}{N \cdot \log N} \quad (2)$$

with N being the number of training samples. These values have proved to guarantee a robust level of privacy in previous studies (Yue et al., 2022).

4.4 Fine-tuning Bio-GPT with DP for Text Generation

To generate the synthetic reports, we fine-tuned BioGPT (generative pre-trained transformer for biomedical text generation) (Luo et al., 2022) on our dataset. BioGPTis a transformer-based

[1] https://transform.england.nhs.uk/information-governance/guidance/artificial-intelligence/

[2] https://docs.ropensci.org/EndoMineR/

sequence-to-sequence model that relies on the GPT-2 architecture and comprises 345 million parameters. BioGPT has been pre-trained on over 15 million PubMed abstracts and has demonstrated increased performance compared to its general domain counterparts for downstream tasks when fine-tuned on biomedical data (Turbitt et al., 2023). We conducted the fine-tuning using the HuggingFace Transformers library (Wolf et al., 2020) along with the Meta AI Opacus library to implement Differential Privacy (Yousefpour et al., 2021). Opacus ensures privacy by applying Differentially Private Stochastic Gradient Descent (DP-SGD), which clips the gradients' L2 norm and adds Gaussian noise to maintain the privacy of the model parameters during the training process.

All the experiments were executed on an NVIDIA DGX server running GNU/Linux 5.4.0-125-generic x86_64, with MLflow integrated into the CSC MLOPs[3] environment to ensure experiment reproducibility, collaboration, and scalability. The specific hyperparameters considered included learning rate, batch size, number of epochs, maximum sequence length, and temperature for the generation process. Fine-tuning was performed with causal language modeling (CLM) objective. The hyperparameter values are described in Table 1.

Table 1: BioGPT fine-tuning hyperparameters

Hyperparameter	**Value**
Batch size per GPU	16
Learning rate	1e-5
Number of training epochs	25
Epsilon ϵ	4

4.5 Generation of DP synthetic endoscopy text reports

4.5.1 Generation process

Control codes were used to steer the generation of specific report types (e.g. OGD, colonoscopy, EUS, ERCP) by the fine-tuned BioGPT model. This technique facilitated the targeted generation of texts according to the different kinds of endoscopic procedures. The input format for this generation process can be conceptualised as: Input = Control Code + Separator + Initial Context.

We built a text generation pipeline, refined through iterative clinician feedback to optimise the authenticity and relevance of the generated reports. The key generation hyperparameters were set as follows:

- **Length Constraints**: The generated reports' maximum length was set to 400 words to reflect the typical lengths of endoscopy reports.

- **Temperature**: This parameter controls the randomness of the generated output by scaling the logits before applying softmax, defined by the equation:

$$P(token) = \frac{\exp(\frac{\log(o_i)}{T})}{\sum_j \exp(\frac{\log(o_j)}{T})} \qquad (3)$$

Here, T represents the temperature, o_i the logits, and $P(token)$ the probability of selecting *token* as the next token. The temperature was set to $T = 0.9$ based on recommendations from domain experts to balance creativity with accuracy.

- **No Repeat Ngram Size**: This parameter was established at 4 to prevent the repetition of any four-word sequence within the generated text, enhancing the uniqueness and readability of the reports.

4.6 Assessment of the Similarity of DP-generated reports

While DP theoretically offers a high level of privacy, its practical effectiveness in safeguarding patient data still requires empirical verification. Recent work has indeed shown that misuses of DP in Deep Learning have often led to limited actual privacy (Blanco-Justicia et al., 2022). As discussed in Section 3, the Hamming and Levensthein distances are often used to assess the privacy of generated reports. However, considering the varying lengths and content complexity of medical reports, these methods may not fully capture the nuances of text similarity, and therefore, may not appropriately assess the privacy of generated reports.

ROUGE-L (Lin, 2004) is a metric which is particularly valuable for evaluating text similarity in generation tasks where structural coherence and order of information are crucial. Unlike BLEU, which focuses on precision by measuring how many words in the generated text appear in the reference texts, ROUGE-L relies on recall, assessing how much of the original report is captured in the generated text. Specifically, ROUGE-L

[3] https://github.com/GSTT-CSC/MLOps

relies on the longest common sub-sequence (LCS) shared between the generated and reference texts, providing a measure of the longest sequence of words appearing in both texts in the same order. ROUGE-L is a normalized metric, therefore making it robust to length variations between the original and generated reports. Our approach to assess the similarity of DP-generated synthetic reports is the following:

1. **Distribution Analysis of ROUGE-L scores**: We compute the ROUGE-L score between each synthetic report and each of the original patient reports. We then keep the highest ROUGE-L score for each of the synthetic reports and compute the resulting distribution. This process is done both for synthetic data generated with and without DP. We then compare DP and non-DP distributions to assess the impact of DP on text similarity and privacy enhancement.
2. **Inclusion of distinctive outliers**: 34 outliers with unique combinations of endoscopic findings are included in the training set. These outliers are text reports of typical length, containing phrasings or combinations of medical conditions not typically found in endoscopy reports. Developed in collaboration with a gastroenterologist, they are distinctive enough that reproducing them directly in synthetic reports could result in patient re-identification.

4.7 Evaluating the Text Quality of DP-generated reports

To evaluate the language quality of generated reports with and without DP, we use the language scoring model introduced in Section 3 (in process of publication), this model takes an individual report as input and assign a score to it based on its language quality. The score ranges from 0 to 1, with higher scores indicating better language quality of the text.

4.8 Evaluating the Utility of DP-generated reports in Downstream Tasks

4.8.1 Baseline Description and Evaluation Metrics

The utility of the generated synthetic reports was evaluated by trying to improve a clinically relevant 4-class classification problem. This involves categorising endoscopy free-text reports based on the length of an endoscopically detectable premalignant lesion: Barrett's Oesophagus (BO) (Fitzgerald et al. (2014); Hameeteman et al. (1989)). The categories are: Long, Short, No Barrett's, and Insufficient, relating to the detection of a long or short segment of BO, a definite lack of detection of BO, or an insufficient description, respectively. The baseline model, detailed in Table 3, is a BERT-based transformer with a linear layer for classification, currently used in clinical practice. It was trained with optimized hyperparameters described in the Appendix (4).

This baseline (Figure 1.a) will be compared against three distinct approaches (Figure 1.b,c,d), as detailed in the subsequent sections (4.8.2, 4.8.3). Given the slightly imbalanced nature of the original training set and the varying clinical relevance of the classes, per-class metrics such as AUC-ROC and F1-Score were recorded. The performance of the baseline and subsequent models was assessed on a test set, using an 80/20 stratified split for each random seed. Results were averaged across three random seeds to ensure robustness.

4.8.2 Synthetic Data Augmentation

The first approach to enhancing the baseline model involved augmenting the training set with 735 DP synthetic gastroscopy reports specifically related to Barrett's Oesophagus. Each report was manually annotated by a domain expert. An overview of the class distributions before synthetic data augmentation is presented in the Appendix (5). The BERT-based classifier was then retrained using the augmented dataset while maintaining the same hyperparameters to allow for a direct comparison of performance changes.

4.8.3 Task-Adaptive Pretraining with Synthetic Data

In the current NLP landscape, LLMs are typically pre-trained on general domain dataset using tasks such as Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) (Devlin et al., 2019; Liu et al., 2019). Although these models exhibit strong performance across various downstream tasks, research has shown that continued pre-training on domain-specific texts can further enhance their effectiveness (Gururangan et al., 2020; Li et al., 2023; Shi et al., 2023; Margatina et al., 2022). In this study, the target domain is gastroenterology text reports. Our second experiment involves task-adaptive pre-training (TAPT) of the baseline model using two separate datasets:

1. Synthetic Data: 10,000 synthetic endoscopy text reports generated with our Differential Privacy pipeline.
2. Real Patient Data: 10,000 endoscopy text reports extracted from the hospital's Electronic Patient Records (EPR) that were not part of the training or evaluation sets of the baseline model. These reports were selected to match the variety and number found in the synthetic dataset.

We used these datasets to conduct domain-adaptive pre-training on the pre-trained BERT model before fine-tuning it for classification tasks and evaluation. Domain-adaptative pre-training was performed using a Masked Language Modeling (MLM) objective. The aim of this experiment is to quantify the utility of synthetic data in comparison to original patient data in terms of enhancing the model's performance on downstream tasks. In both cases, TAPT experiments were conducted using the hyperparameters listed in Table (2).

Table 2: Hyperparameters for TAPT Experiments

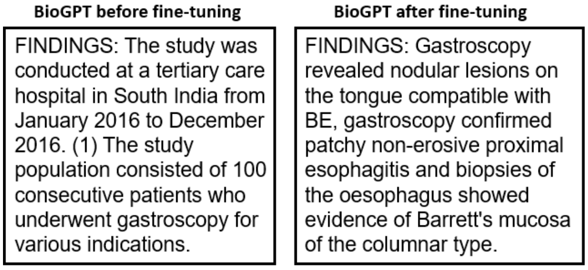
Hyperparameter	Value
Number of Pretraining Epochs	100
Number of Fine-tuning Epochs	8
Pretraining Learning Rate	1e-4
Fine-tuning Learning Rate	6.85e-5
Warming Up Steps	1000

5 Results

5.1 Generation of DP Synthetic Reports

10,000 DP synthetic reports were generated using the input defined in 4.5. We compare the output of the model, before and after fine-tuning, as depicted in Figure 2.

We observe that before fine-tuning, the output of the model resembles a PubMed abstract that mentions the report type, gastroscopy in this case. However, it appears like a study rather than an individual patient's endoscopy report. After fine-tuning, the model's output presents as a well-formatted gastroscopy report, with findings related to Barrett's Oesophagus.


BioGPT before fine-tuning

FINDINGS: The study was conducted at a tertiary care hospital in South India from January 2016 to December 2016. (1) The study population consisted of 100 consecutive patients who underwent gastroscopy for various indications.

BioGPT after fine-tuning

FINDINGS: Gastroscopy revealed nodular lesions on the tongue compatible with BE, gastroscopy confirmed patchy non-erosive proximal esophagitis and biopsies of the oesophagus showed evidence of Barrett's mucosa of the columnar type.

Figure 2: Comparison of BioGPT output for the generation of synthetic reports, before (left) and after (right) fine-tuning.

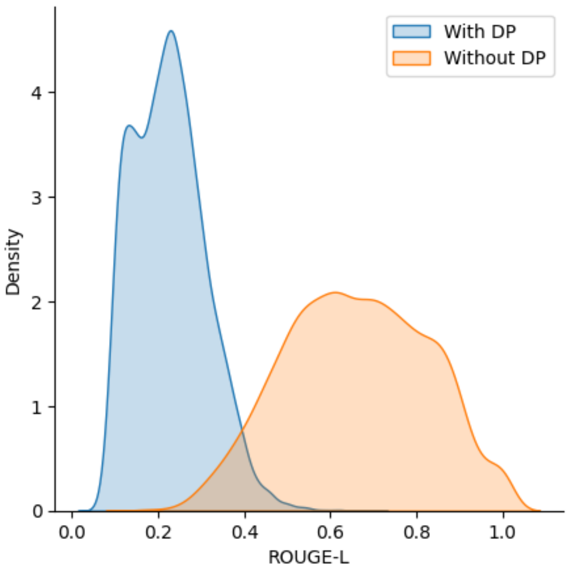


Figure 3: Distributions of maximum ROUGE-L scores between original and synthetic reports, with (left) and without (right) Differential Privacy.

5.2 Text Similarity Analysis

The distributions of maximum ROUGE-L scores between original and synthetic reports (both DP and non DP) are depicted in Figure 3. We observe a significant shift in distribution between the synthetic reports generated with Differential Privacy compared to those generated without it. The ROUGE-L scores for the DP-generated reports span from 0.058 to 0.690, with an average of 0.226, indicating that the DP-generated reports significantly differ from the training set. In contrast, the ROUGE-L scores for the non-DP-generated reports span from 0.165 to 1.0, with an average of 0.660, indicating that some generated reports are highly similar to the training set.

After careful review of the synthetic reports generated with and without Differential Privacy (DP) with a domain expert, we confirmed that no outlier was directly regenerated in either the synthetic DP or non-DP reports.

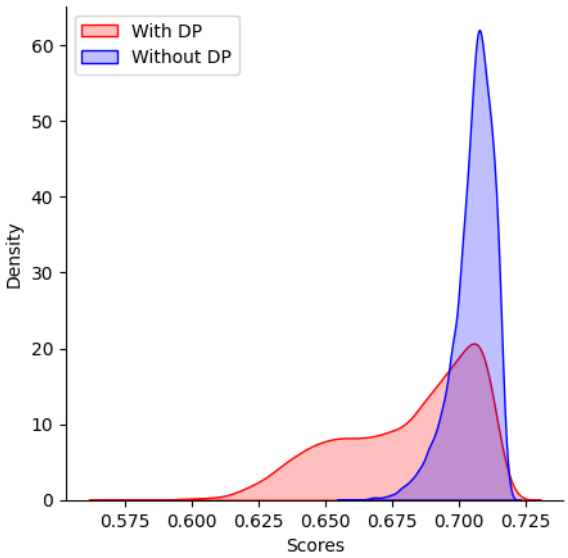


Figure 4: Distributions of synthetic data language quality scores with (left) and without (right) Differential Privacy.

5.3 Language Quality Evaluation

Figure 4 shows that synthetic reports generated without DP exhibit language scores centred around higher values. While synthetic reports generated with DP have a similar tendency, their scores are more distributed toward the lower end, resulting in a broader, shorter-tailed distribution. Despite this variation, the overall spread remains relatively constrained, indicating a slight reduction in the text quality of reports generated with DP.

5.4 Utility Evaluation

The results of the three utility evaluation experiments are summarized in Table 3. The optimized baseline model already achieves high performance across all classes, with the ‘Long’ class showing the highest average F1-score (0.958) and the ‘Insufficient’ class the lowest (0.822).

The most notable conclusion from these experiments is that task-adaptive pretraining considerably improves the baseline performance for all classes, especially for the ‘Insufficient’ class, which sees an F1-score increase of 0.089. The ‘Long’ class, which already performed well, also shows an improvement of 0.022.

The primary goal of this paper is to assess the utility of synthetic data generated with differential privacy. As expected, the baseline improvement using DP synthetic data is not as significant as with real patient data, likely due to the set privacy level (epsilon = 4). However, TAPT using synthetic data with DP still enhances the F1 scores across all classes, with the ‘Insufficient’ class showing the most significant improvement of 0.034. The Long class, despite its high performance, also showed an improvement while performing TAPT with DP synthetic data, with an F1 score improvement of 0.003.

Data augmentation with labelled synthetic DP text reports also improved performance across most classes, though results were more inconsistent. This variability may be due to the limited number of additional annotated reports, as the annotation process is time-consuming and constrained by a shortage of expert annotators.

6 Discussion and Conclusion

We have fine-tuned a pre-trained large language model with Differential Privacy to generate privacy-preserved synthetic endoscopy reports. We leveraged a highly specific in-house training set of over 90,000 endoscopy free-text reports. Using our pipeline, we generated a set of 10,000 diverse synthetic endoscopy reports, available for further research on a per-query basis. The utility of the synthetic reports was assessed by attempting to improve a clinically useful high performing classification baseline. The synthetic reports were used to augment the training set of the baseline and to pretrain the baseline classifier using a task-adaptive pretraining framework. A pre-training experiment with real patient data was also conducted for direct comparison.

Table 3 demonstrates that DP-generated synthetic data can significantly improve the performance of a real-world downstream task. Specifically, our study demonstrates the superiority of TAPT methods. It is important to highlight that, in comparison to supervised-learning approaches, TAPT does not require additional labeled data points, which significantly reduces the need for data annotation resources, a primary bottleneck in developing robust supervised learning models. However, the best-performing model remains the one pre-trained with real patient data.

The privacy preservation of DP-generated reports is quantified by assessing their similarity to the original data compared to that of synthetic data generated without DP. We observed an important difference in ROUGE-L scores between the DP and non-DP generated synthetic reports. Therefore, we can conclude that, in our study, DP effectively pre-

Table 3: Comparison of model performance across different approaches, including the baseline BERT-based model (a), synthetic data augmentation with 735 differentially private reports (b), task-adaptive pretraining with 10,000 real patient reports (c), and task-adaptive pretraining with 10,000 differentially private synthetic reports (d).

Approach	Long		No Barretts		Short		Insufficient	
	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1
(a) Baseline	$0.988_{0.008}$	$0.958_{0.005}$	$0.992_{0.005}$	$0.924_{0.020}$	$0.984_{0.012}$	$0.925_{0.013}$	$0.979_{0.013}$	$0.822_{0.034}$
(b) Data augmentation	$0.997_{0.001}$	$0.961_{0.016}$	$0.982_{0.005}$	$0.914_{0.017}$	$0.987_{0.004}$	$0.944_{0.010}$	$0.967_{0.031}$	$0.798_{0.029}$
(c) TAPT with real data	$0.997_{0.002}$	$0.980_{0.003}$	$0.997_{0.002}$	$0.961_{0.014}$	$0.994_{0.002}$	$0.967_{0.010}$	$0.988_{0.004}$	$0.911_{0.022}$
(d) TAPT with DP synthetic	$0.991_{0.005}$	$0.961_{0.025}$	$0.989_{0.006}$	$0.935_{0.025}$	$0.987_{0.007}$	$0.945_{0.024}$	$0.980_{0.005}$	$0.856_{0.053}$

vents the replication of sensitive training samples. A set of outliers was also introduced in the training set, and we concluded that unique clinical findings could not be regenerated by the models.

6.1 Limitations and Future Work

While this study aims to explore various methods to assess the utility and privacy of DP-generated endoscopy text reports, we acknowledge several limitations. First, we used a fixed value of the privacy parameter epsilon throughout this study. Future work should assess the impact of epsilon on the privacy-utility trade-off.

We have compared the performance of a classification baseline to several approaches leveraging generated synthetic reports (Table 3), but we have not compared the baseline to a classifier solely trained on synthetic data due to the limited availability of high-quality annotation resources. Moreover, future work should also consider comparing our approach with other existing methods of synthetic data generation and privacy protection to provide a more comprehensive evaluation.

It is also important to highlight that the generative models were fine-tuned on endoscopy data from a London hospital, which might introduce population bias or an over-representation of specific endoscopic conditions due to the local context.

In this study, we assessed the text quality of the generated text; however, no evaluation of clinical relevance was conducted. Clinical accuracy is essential for specific downstream tasks as it ensures the medical reliability of the generated reports and prevents confusion. Users should remain mindful of this aspect when using our synthetic reports.

7 Acknowledgements

We would like to express our gratitude to the Clinical Scientific Department at Guy's and St' Thomas' NHS Foundation Trust for providing us with compute resources.

8 Availability of Generated Reports and Code

The generated synthetic endoscopy text reports, along with the corresponding code used for their generation, are made available on a per-request basis.

References

2023. Nhs national data opt-out.

Mahdi Abdollahi, Xiaoying Gao, Yi Mei, Shameek Ghosh, Jinyan Li, and Michael Narag. 2021. Substituting clinical features using synthetic medical phrases: Medical text data augmentation techniques. *Artificial Intelligence in Medicine*, 120:102167.

Edmon Begoli, Kris Brown, Sudarshan Srinivas, and Suzanne Tamang. 2018. Synthnotes: A generator framework for high-volume, high-fidelity synthetic mental health notes. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 951–958.

Alberto Blanco-Justicia, David Sánchez, Josep Domingo-Ferrer, and Krishnamurty Muralidhar. 2022. A critical review on the use (and misuse) of differential privacy in machine learning. *ACM Comput. Surv.*, 55(8).

Roxana Daneshjou, Mary P. Smith, Mary D. Sun, Veronica Rotemberg, and James Zou. 2021. Lack of transparency and potential bias in artificial intelligence data sets and algorithms. *JAMA Dermatology*, 157(11):1362.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Preprint*, arXiv:1810.04805.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. *Calibrating Noise to Sensitivity in Private Data Analysis*, page 265–284. Springer Berlin Heidelberg.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*.

Rebecca C Fitzgerald, Massimiliano di Pietro, Krish Ragunath, Yeng Ang, Jin-Yong Kang, Peter Watson, Nigel Trudgill, Praful Patel, Philip V Kaye, Scott Sanders, Maria O'Donovan, Elizabeth Bird-Lieberman, Pradeep Bhandari, Janusz A Jankowski, Stephen Attwood, Simon L Parsons, Duncan Loft, Jesper Lagergren, Paul Moayyedi, Georgios Lyratzopoulos, John de Caestecker, and British Society of Gastroenterology. 2014. British society of gastroenterology guidelines on the diagnosis and management of barrett's oesophagus. *Gut*, 63(1):7–42.

Ghadeer O. Ghosheh, Jin Li, and Tingting Zhu. 2024. A survey of generative adversarial networks for synthesizing structured electronic health records. *ACM Computing Surveys*, 56(6):1–34.

Jiaqi Guan, Runzhe Li, Sheng Yu, and Xuegong Zhang. 2018. Generation of synthetic electronic medical record text. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 374–380.

Jiaqi Guan, Runzhe Li, Sheng Yu, and Xuegong Zhang. 2021. A method for generating synthetic electronic medical record text. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 18(1):173–182.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *Preprint*, arXiv:2004.10964.

W Hameeteman, GNJ Tytgat, HJ Houthoff, and JG Van Den Tweel. 1989. Barrett's esophagus; development of dysplasia and adenocarcinoma. *Gastroenterology*, 96(5):1249–1256.

Julia Ive, Natalia Viani, Joyce Kam, Lucia Yin, Somain Verma, Stephen Puntis, Rudolf N. Cardinal, Angus Roberts, Robert Stewart, and Sumithra Velupillai. 2020. Generation and evaluation of artificial mental health records for natural language processing. *npj Digital Medicine*, 3(1).

Oleksandra Klymenko, Stephen Meisenbacher, and Florian Matthes. 2022. Differential privacy in natural language processing the story so far. In *Proceedings of the Fourth Workshop on Privacy in Natural Language Processing*, pages 1–11, Seattle, United States. Association for Computational Linguistics.

Jianfu Li, Yujia Zhou, Xiaoqian Jiang, Karthik Natarajan, Serguei Vs Pakhomov, Hongfang Liu, and Hua Xu. 2021. Are synthetic clinical notes useful for real natural language processing tasks: A case study on clinical entity recognition. *Journal of the American Medical Informatics Association*, 28(10):2193–2201.

Shiyang Li, Semih Yavuz, Wenhu Chen, and Xifeng Yan. 2023. Task-adaptive pre-training and self-training are complementary for natural language understanding. *Preprint*, arXiv:2109.06466.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *Preprint*, arXiv:1907.11692.

Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. 2022. BioGPT: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6).

Katerina Margatina, Loic Barrault, and Nikolaos Aletras. 2022. On the importance of effectively adapting pretrained language models for active learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 825–836, Dublin, Ireland. Association for Computational Linguistics.

Oren Melamud and Chaitanya Shivade. 2019. Towards automatic generation of shareable synthetic clinical notes using neural language models. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 35–45, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Trishan Panch, Heather Mattie, and Leo Anthony Celi. 2019. The "inconvenient truth" about AI in healthcare. *npj Digital Medicine*, 2(1).

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: Measuring the gap between neural text and human text using divergence frontiers. *Advances in Neural Information Processing Systems*, 34:4816–4828.

Atiquer Rahman Sarkar, Yao-Shun Chuang, Noman Mohammed, and Xiaoqian Jiang. 2024. De-identification is not always enough. *Preprint*, arXiv:2402.00179.

Zhengxiang Shi, Francesco Tonolini, Nikolaos Aletras, Emine Yilmaz, Gabriella Kazai, and Yunlong Jiao. 2023. Rethinking semi-supervised learning with language models. *Preprint*, arXiv:2305.13002.

Ruixiang Tang, Xiaotian Han, Xiaoqian Jiang, and Xia Hu. 2023. Does synthetic data generation of llms help clinical text mining? *Preprint*, arXiv:2303.04360.

Oisn Turbitt, Robert Bevan, and Mouhamad Aboshokor. 2023. Mdc at biolaysumm task 1: Evaluating gpt models for biomedical lay summarization. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Honghan Wu, Minhong Wang, Jinge Wu, Farah Francis, Yun-Hsuan Chang, Alex Shavick, Hang Dong, Michael T. C. Poon, Natalie Fitzpatrick, Adam P. Levine, Luke T. Slater, Alex Handy, Andreas Karwath, Georgios V. Gkoutos, Claude Chelala, Anoop Dinesh Shah, Robert Stewart, Nigel Collier, Beatrice Alex, William Whiteley, Cathie Sudlow, Angus Roberts, and Richard J. B. Dobson. 2022. A survey on clinical natural language processing in the united kingdom from 2007 to 2022. *npj Digital Medicine*, 5(1).

Chao Yan, Ziqi Zhang, Steve Nyemba, and Bradley A Malin. 2021. Generating electronic health records with multiple data types and constraints.

Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, Graham Cormode, and Ilya Mironov. 2021. Opacus: User-friendly differential privacy library in PyTorch. *arXiv preprint arXiv:2109.12298*.

Xiang Yue, Huseyin A. Inan, Xuechen Li, Girish Kumar, Julia McAnallen, Hoda Shajari, Huan Sun, David Levitan, and Robert Sim. 2022. Synthetic text generation with differential privacy: A simple and practical recipe. *arXiv preprint*.

Sebastian S. Zeki. 2018. Endominer for the extraction of endoscopic and associated pathology data from medical reports. *Journal of Open Source Software*, 3(24):701.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with BERT. *CoRR*, abs/1904.09675.

Ziqi Zhang, Chao Yan, Thomas A Lasko, Jimeng Sun, and Bradley A Malin. 2020. Synteg: a framework for temporal structured electronic health data simulation. *Journal of the American Medical Informatics Association*, 28(3):596–604.

Yongxin Zhou, Fabien Ringeval, and François Portet. 2023. A survey of evaluation methods of generated medical textual reports. In *Proceedings of the 5th Clinical Natural Language Processing Workshop*, pages 447–459, Toronto, Canada. Association for Computational Linguistics.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 1097–1100.

A Classification baseline details

Table 4: Classification Baseline Hyperparameters

Hyperparameter	Search Space	Baseline model value
Batch size	{16, 32, 64}	32
Learning rate	[1e-6, 1e-3]	6.85e-5
Number of training epochs	[1, 10]	8
Warming steps fraction	[0.1, 0.5]	0.4

Table 5: Class Distributions Before and After Data Augmentation

Class	Before Augmentation	After Augmentation
Insufficient	279	475
Long	1,649	1,688
Short	1,901	1,901
No Barrett's	288	788
Total Reports	4,117	4,852

Evaluating the Robustness of Adverse Drug Event Classification Models Using Templates

Dorothea MacPhail[1], **David Harbecke**[1], **Lisa Raithel**[1,2,3], **Sebastian Möller**[1,2]
[1]German Research Center for Artificial Intelligence (DFKI), Berlin
[2]Quality & Usability Lab, Technische Universität Berlin
[3]BIFOLD – Berlin Institute for the Foundations of Learning and Data
[1]{firstname}.{lastname}@dfki.de

Abstract

An adverse drug effect (ADE) is any harmful event resulting from medical drug treatment. Despite their importance, ADEs are often under-reported in official channels. Some research has therefore turned to detecting discussions of ADEs in social media. Impressive results have been achieved in various attempts to detect ADEs. In a high-stakes domain such as medicine, however, an in-depth evaluation of a model's abilities is crucial. We address the issue of thorough performance evaluation in English-language ADE detection with handcrafted templates for four capabilities: Temporal order, negation, sentiment, and beneficial effect. We find that models with similar performance on held-out test sets have varying results on these capabilities.

1 Introduction

When a trained model is applied to real-world data, it may be confronted with phenomena that are under-represented or non-existent in the training data (Belinkov and Bisk, 2019; Moradi and Samwald, 2022). This raises the question of how to evaluate a model's performance and generalization abilities. Reporting summary statistics and held-out test set performance is a common practice in model evaluation. While this can provide an indication of the model's performance and ability to generalize, there are some issues with this practice. Firstly, held-out test sets often arise from the same distribution as the training data and will, therefore, exhibit the same patterns and biases to a high degree. Real-world data, however, may have different feature distribution or exhibit noise. Held-out testing, therefore, often provides an unsatisfactory estimation of a model's performance and generalization abilities (Belinkov and Bisk, 2019; McCoy et al., 2019; Ribeiro et al., 2018).

Secondly, a high model score does not necessarily reveal what the model has learned during training. Research has shown that a model may not learn relevant patterns but instead base its decisions on shallow heuristics or proxies (McCoy et al., 2019). Benchmark challenges have attempted to address this issue by testing models on a wide range of aspects of language (Wang et al., 2019). However, not all aspects can be tested in a benchmark, and the benchmark itself may exhibit unintended biases (Kiela et al., 2021), so the question of what a model has learned remains.

Inspired by the behavioral testing suite CheckList (Ribeiro et al., 2020), we propose the use of template-based test cases to test different capabilities of adverse drug effect (ADE) classification models. ADEs are any harmful consequence to a patient due to medical drug intake. Due to the potential detrimental outcomes of ADEs, the detection of ADEs is an important goal in health-related NLP and has been a subject of research for a considerable time. We test models in understanding of temporal order, positive sentiment, beneficial effects and negation (see Table 1).

In high-stakes domains such as medicine, an in-depth evaluation of a model's abilities is crucial. Related work (Section 2), however, suggests that shortcomings towards selected linguistic phenomena and reliance on proxies for model decisions may exist in models in the biomedical domain.

In this work[1], two transformer-based models for the detection of ADEs in user reports on social media were fine-tuned and tested by conventional held-out testing as well as additional template-based tests. The results of held-out testing and the template-based tests were compared in order to better understand (i) the models' shortcomings and (ii) the potential gaps in knowledge that can occur when a model's abilities are only evaluated via test set performance. We find that models underper-

[1] The templates and code can be found at https://github.com/dfki-nlp/ade_templates

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 25–38
August 16, 2024. ©2024 Association for Computational Linguistics

Test Name	Label	Test Description	Example Test Case
Temporal Order standard	no ADE	ADE occurs before drug intake	Before taking cymbalta, I experienced Insomnia.
	ADE	ADE occurs after drug intake	Before having acid reflux, I was put on zoloft.
Temporal Order single time entity	no ADE	ADE occurs before drug intake expressed by a time entity	I was experiencing bad pain in my right arm for 2 weeks, now I started being medicated with Effexor XR.
	ADE	ADE occurs after drug intake expressed by a time entity	3 months ago I started being treated with zoloft, now I started encountering excellerated heart rate.
Temporal Order double time entities	no ADE	ADE occurs before drug intake expressed by two related time entities	3 weeks ago I started suffering from bad pain in my right arm, I have been taking effexor for 2 days.
	ADE	ADE occurs after drug intake expressed by two related time entities	I was enduring Insomnia for 6 weeks, 8 weeks ago I started taking cymbalta.
Positive Sentiment	ADE	ADE occurrence is reported with positive sentiment	I'm taking cymbalta and experiencing cravings for sweets. Still, I am happy my symptoms have reduced.
Beneficial Effect	no ADE	Secondary effect of a drug that is beneficial to the patient	I'm taking Effexor XR and experiencing weight loss. I'm happy because I was trying to lose weight anyway.
	ADE	Secondary effect of a drug that is an ADE as it is not beneficial	For me, weight loss is a side-effect of effexor. It's a problem because I am already underweight.
Negation	no ADE	ADE is negated	I am taking zoloft without suffering from acid reflux.
	ADE	Statement contains negation, ADE is not negated	That's not true, I took zoloft and encountered Insomnia.

Table 1: Overview of all four capabilities tested with example test cases. The temporal order capability has three variations. All test cases have an assigned label, either ADE or no ADE. Filled-in entities are underlined in the example test cases. All test cases are hand-crafted.

form on some capabilities and show differences in some capabilities *despite highly similar F_1-scores on the held-out test set*. We therefore provide the following contributions:

- A curated test bench of 99 templates with 1505 variations to investigate the robustness of ADE classification models across four capabilities.
- A comparison of two popular transformer-based models on long-tail linguistic phenomena in the classification of ADEs.

2 Related Work

Studies on the detection of ADEs in user-generated texts have been conducted since approximately 2010, when Leaman et al. published the first English dataset within this domain. The usual downstream tasks are those common in information extraction: Document classification, to find relevant documents containing mentions of adverse effects; named entity recognition, to identify medication and disease-related mentions; and relation classification, to establish associations between the entity mentions. Approaches for all of these tasks range from rule- and lexicon-based systems (Leaman et al., 2010; Nikfarjam and Gonzalez, 2011) to traditional machine learning pipelines (Gurulingappa et al., 2012; Ginn et al., 2014; Segura-Bedmar et al., 2014) and, recently, deep neural networks (Huynh et al., 2016), specifically transformer-based setups (Weissenbacher et al., 2019; Miftahutdinov et al., 2020; Gusev et al., 2020; Magge et al., 2021b).

However, even advanced models struggle with the supposedly simple task of classifying a document into either "contains an ADE" (henceforth ADE) or "does not contain an ADE" (no ADE), a standard binary classification that is still necessary to find relevant documents for further information extraction. This is often due to a strong class imbalance (in most cases, the documents containing ADEs are in the minority), the usual noise in social media data, ambiguities in health-related statements of patients, and general weaknesses of language models in coping with certain linguistic phenomena not only with respect to ADEs.

For example, Scaboro et al. (2021) have studied the extraction of ADEs from tweets using BERT, SpanBERT (Joshi et al., 2020), and PubMedBERT (Gu et al., 2021). They tested all three models' ability to handle negation and detect shortcomings in all three models. Moradi and Samwald (2022) investigated the robustness of four transformer models specialized in the biomedical and clinical domain over a variety of tasks such as sentence classification, inference, and question answering. The models' robustness is tested by adding minor meaning-preserving changes to the input with the goal of fooling the model. Their findings highlight the vulnerability of state-of-the-art transformer-based models to adversarial input.

Finally, there is CheckList (Ribeiro et al., 2020), a model-agnostic framework aimed at testing a trained model's behavior and gaining an in-depth understanding of its potential shortcomings. CheckList guides the creation of test cases based on natural language *capabilities*, which are used as new inputs to the trained model and subsequently evaluated. The idea is to determine which capabilities (e.g., negation handling, robustness) are necessary for the task the model is intended to perform. Ribeiro et al. (2020) identify three possible test types which can be used for testing the capabilities: the Minimum Functionality Test (MFT), which targets a specific behavior similar to a unit test; the Invariance Test (INV), where the model's robustness to irrelevant perturbations is tested; and the Directional Expectation Test (DIR), which consists of adding perturbations that are expected to lead to a specific outcome. Ribeiro et al. (2020) observe that the CheckList-based evaluation approach could not only uncover bugs in previously tested models but also that CheckList can make the search for bugs more systematic. Recently, updates to CheckList, AdaTest (Ribeiro and Lundberg, 2022) and AdaTest++ (Rastogi et al., 2023), were proposed which assist the user in finding bugs by suggesting topics and test cases in a semi-automated process. While these are valuable additions, we decided to use the template-based approach for this project because we had pre-selected capabilities that we wanted to test with full control over the template design.

CheckList applications include the evaluation of general capabilities of models (Xie et al., 2021) as well as evaluating models in specialized tasks such as offensive speech detection (Bhatt et al., 2021; Manerba and Tonelli, 2021) and automatic text simplification (Cumbicus-Pineda et al., 2021). For the specialized tasks, the authors use CheckList to guide their testing approach by defining new capabilities specific to the task at hand.

In the biomedical and clinical domain, Ahsan et al. (2021) use CheckList to test four linguistic capabilities (negation, temporal order, misspellings, and attributions) on their transformer-based model with a dataset of clinical discharge notes. One of their findings is that the model struggles to correctly distinguish between past and present mentions of substance use in the discharge notes. The detection of ADEs, however, is not part of the research.

The exposure of potential weaknesses in transformer-based models in the biomedical domain motivates an in-depth analysis of models used for ADE detection. To our knowledge, a systematic template-based approach to test model capabilities has not yet been applied to ADE detection.

3 Methods

We use templates to test a selection of linguistic capabilities of binary ADE classification models. To this end, we first manually create templates (see Section 3.1) and then sets of test cases, by using entities to fill placeholders in the templates (see Section 3.3). We then evaluate two fine-tuned classification models on these test cases and compare their predictions with each other and with the models' performance on the held-out test set.

Example 1: Template for Temporal Order (`ADE`)

I started taking `{drug}` before I experienced `{ade}`.

We test four capabilities: *Temporal Order*, *Positive Sentiment*, *Beneficial Effect*, and *Negation* (see Section 3.2). 99 base templates are created with 1505 variations (for details see Table 5 in Appendix A). Each template is also assigned a label (`ADE`/`no ADE`) in accordance with published guidelines for the annotation of ADEs (see Section 4.1.1). The template in Example 1 provides a test case for the capability *Temporal Order* and has a positive label (`ADE`). Filled-in template examples for every capability we test are listed in Table 1. The filled-in templates (test cases) serve as the input to the fine-tuned model for inference. In the following, we present more details about the template creation and the investigated capabilities.

3.1 Template Creation

Template-based evaluation is most effective with a large number of test cases that cover a diverse range of potential inputs. These test cases are based on templates, which include placeholders. For every placeholder, there is a list of potential entity fill-ins as in Example 1, `{drug}` and `{ade}`, which could be filled with, e.g., *Effexor* and *nausea*. The abstraction of test cases to templates allows to systematically capture important linguistic scenarios while creating a large number of different test cases. The process is visualized in Figure 1.

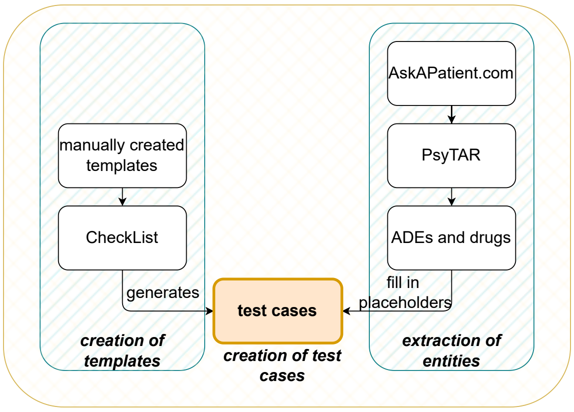


Figure 1: The process for creating test cases.

In the interest of linguistic diversity, variations of base templates were introduced for all capabilities except *Beneficial Effect*. For *Temporal Order* and *Negation* templates, the vocabulary of the base template was modified to increase diversity. *Positive Sentiment* templates underwent syntax variations by exchanging or removing the conjunction between the two phrases.

The templates have a mean token count of 10.6 and 13.4 for the `no ADE` and `ADE` class respectively[2]. After filling in the entities for the placeholders, the average test case length in the experiments is 14.7 for the `no ADE` class and 16.6 for the `ADE` class.

3.2 Capabilities

The choice of capabilities for this work is inspired by considerations on abilities a robust ADE classification model should possess and shortcomings of biomedical models as reported in Section 2. We based the phrasing of the templates on linguistic properties of social media posts: First-person usage, mostly single short sentences, and colloquial language. Contractions were used occasionally. However, usernames, misspellings, and non-standard grammar and punctuation were not applied in the templates as they manifest a separate capability. All templates created can be viewed as templates for a CheckList Minimum Functionality Test (Ribeiro et al., 2020).

To verify the existence of the described phenomena in the dataset, we randomly sampled 1,000 documents and let two annotators check each tweet for the occurrence of these phenomena. The annotations showed that eight of the sampled tweets contain expressions of temporal order, one positive sentiment, one beneficial effect, and one negation. This sample showed that, as expected, the phenomena are rather rare but still exist in the long tail of the data distribution. Nevertheless, an expert would expect a good classification model to have these capabilities.

Temporal Order The templates for testing *Temporal Order* adapt the temporal structure test of Ribeiro et al. (2020) and investigate the model's ability to correctly process information on past, present, and future as expressed in text. In the context of ADE detection, it is important for the model to "understand" temporal order since an effect cannot be an ADE if it occurred before the drug intake. According to the annotation guidelines based on which the data we use for fine-tuning was annotated, an effect occurring after a drug intake was labeled as ADE if the patient draws a connection between the effect and drug intake. Therefore, the templates assume an ADE when a harmful effect occurs after the drug intake.

Positive Sentiment ADEs are often reported using negative sentiment (Alhuzali and Ananiadou, 2019). If many ADE reports contain negative sentiment, an ADE detection model might perform well by using negative sentiment as a proxy. Nevertheless, a report might also be expressed favorably. This could be the case when a patient experiences relief from the original symptoms alongside a mild ADE. Therefore, an ADE detection model should recognize ADEs even when expressed in a positive framing so as not to miss out on less severe ADEs.

Beneficial Effects The third capability is the correct distinction between ADEs and beneficial effects. The latter are secondary effects of a drug that are not related to the reason for using the med-

[2] Tokens were split at whitespace.

ication and which have, nevertheless, a positive outcome for the patient. Note that an effect may be regarded as positive or negative depending on the patient, their general health, and the context. Weight loss, for instance, may be considered a negative secondary drug effect or a beneficial effect depending on the patient. The tests in this work assume that a positive secondary effect is a beneficial effect, not an ADE. The *Beneficial Effect* test that expects a negative class label (`no ADE`) expresses the occurrence of a beneficial effect. The positive class (`ADE`) test consists of test cases that express an ADE that could be classified as a beneficial effect, but the context states that the user views the effect as negative.

Negation *Negation* templates test the model's ability to process negation in text. Negation detection is a general challenge in NLP and a common phenomenon in language (Hossain et al., 2022; Truong et al., 2022). Thus, it is also an important capability for ADE detection. The *Negation* test that expects a negative class label (`no ADE`) contains a negated ADE. The positive class (`ADE`) test cases include an ADE mention as well as a negation without negating the ADE.

3.3 Entity Placeholders

All templates have entity placeholders for a drug name. Templates for *Temporal Order*, *Positive Sentiment*, and *Negation* also have a placeholder for an ADE entity. Templates for *Beneficial Effect* contain an effect that may be considered an ADE or a beneficial effect depending on the context. A list of the effects used in the *Beneficial Effects* tests is provided in Appendix A.2. Template variations of the *Temporal Order* capability that use time entities have placeholders for time expressions. The placeholders are filled with the respective time expressions from a self-created list of entities.

4 Experiments

We frame ADE detection as a binary classification task. We first describe the experiments on the custom dataset and then the experiments on our template-based test cases.

4.1 Fine-Tuning Experiments

The following describes data, training and evaluation on the custom dataset.

Dataset	#Tweets	ADE Ratio (%)
SMM4H'21 Task 1a	17,426	7.39
SMM4H'17 Task 1	14,880	8.72
NADE	246	0.00
Merged Dataset	**28,468**	**8.75**

Table 2: The number of tweets per dataset and the respective ADE ratio (number of positive samples) of the merged dataset and its three components. 4084 duplicates were removed after merging.

4.1.1 Data

The custom dataset for our experiments consists of three social media corpora: The SMM4H-2021 Shared Task 1a training data (Magge et al., 2021a) (61% of the custom dataset), the SMM4H-2017 Shared Task dataset (Sarker et al., 2018) (38%), and artificially negated tweets from the NADE dataset (Scaboro et al., 2021) (1%), resulting in 28,468 tweets. The data flow and their origin are shown in Figure 2. Dataset statistics are covered in Table 2. In the user-reported texts, each sample either describes an ADE (`ADE`) or does not contain an ADE mention (`no ADE`).

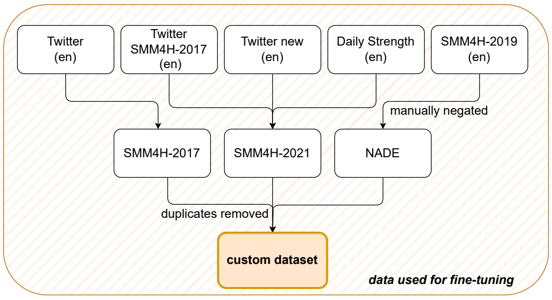


Figure 2: The different data sources for creating the custom dataset for fine-tuning the models.

The SMM4H-2021 Shared Task 1a training data (Magge et al., 2021a) itself consists of posts from Twitter and DailyStrength[3] collected using a list of 81 drugs widespread on the US market (Nikfarjam et al., 2015). The data was annotated by two expert annotators. The annotators did not include beneficial effects in the ADE definition. It further includes some data previously used in the SMM4H-2017 Shared Task (Sarker et al., 2018).

The SMM4H-2017 Shared Task data was collected from Twitter using generic drug names with a total of 250 keywords and subsequently annotated by two annotators. Again, the annotators excluded

[3] www.dailystrength.org

beneficial effects from the ADE definition. Overlapping texts between the SMM4H-2021 data and the SMM4H-2017 data used for our merged custom dataset were removed.

The last part of our custom dataset are artificially negated tweets from the NADE dataset (Scaboro et al., 2021). This dataset consists of tweets originating from the SMM4H-2019 Shared Task (Weissenbacher et al., 2019) and manually negated by annotators. Each negated tweet contains a statement that negates the presence of an ADE. The three components are shown again in Table 2.

We use this merged version of multiple datasets to give the fine-tuning models the best chance to learn different capabilities from varied data. The texts in the custom dataset are between 1 and 34 tokens long.[4] Negative (no ADE) samples are slightly shorter on average (16.2 tokens) than positive (ADE) samples (18.4 tokens). These are slightly longer than our test cases with an average length of 14.7 tokens and 16.6 tokens. Data splits for training, validation, and testing were created with a 70-10-20 ratio and stratified sampling by class label.

4.1.2 Model Fine-Tuning

For the task of ADE classification, we fine-tune BioRedditBERT (Basaldella et al., 2020) and XLM-RoBERTa (Conneau et al., 2020) on the custom dataset described in Section 4.1.1. BioRedditBERT is a BERT-base uncased model related to BioBERT (Lee et al., 2019), a model pre-trained on the original BERT training corpus (English Wikipedia + BookCorpus) as well as on medical texts sourced from PubMed and PMC. It was then further fine-tuned on a corpus of health-related Reddit posts. XLM-RoBERTa is a popular multilingual model with no specific medical pre-training data. We chose these models to gain insights on robustness of a language model with medical knowledge compared with an general domain language model that has no specific medical knowledge.

The inputs were sampled with replacement weighted by class ratio due to the class imbalance. This sampling strategy resulted in a better F_1-score on the validation dataset.

4.1.3 Held-Out Test Set Evaluation

We evaluate the fine-tuned models on the test set using precision, recall, and F_1-score for each class. The main metric we focused on is F_1 of the positive class due to the large class imbalance. This metric was also used for hyperparameter tuning on the validation set. We compare per-class recall to the models' performances on each capability of the test cases. The goal of this comparison is to determine whether the template-based evaluation approach contradicts the overall impression of the model performance measured by held-out test set performance.

Test	Label	#Test Cases
Temporal Order standard	no ADE ADE	1,050 900
Temporal Order single time entity	no ADE ADE	1,050 1,050
Temporal Order double time entities	no ADE ADE	1,575 1,575
Positive Sentiment	ADE	2,700
Beneficial Effect	no ADE ADE	120 120
Negation	no ADE ADE	825 300
Total		**11,265**

Table 3: Number of test cases run per test. We have at least 120 test cases for each capability, so that we can expect our results to be representative.

4.2 Test Case Experiments

We use all templates for each test and randomly select only one template variation per base template for the capabilities *Temporal Order*, *Positive Sentiment*, and *Negation* to have a manageable number of test cases. We created a total of 11,265 test cases, of which 4,620 test cases belong to the negative class (no ADE) and 6,645 belong to the positive class (ADE). Table 3 shows the number of test cases run per test.

A random sample of 15 ADEs, 15 mild ADEs, 5 drug names, 7 single time entities, and 7 relational time entities was taken. A list of sampled ADEs, mild ADEs, and drug names can be viewed in Appendix B.

4.2.1 Drug and ADE Template Fill-Ins

We need expressions of ADEs and medical drugs to fill in the placeholders in the templates. These are automatically extracted from the PsyTAR dataset (Zolnoori et al., 2019) of patient reports on psychiatric medications. The dataset consists of 891 Ask-a-Patient[5] patient forum posts on the topic of four psychiatric medications: Zoloft, Lexapro,

[4] Tokens were split at white spaces.

[5] www.askapatient.com

Cymbalta, and Effexor XR. The corpus was annotated for ADE mentions by four annotators with a health-related background. A mention was considered an ADE “if there is an explicit report of any sign/symptom that the patient explicitly associated them with the drug consumption” (Zolnoori et al., 2019). All four drug names of PsyTAR were extracted as well as two spelling variations of “Effexor XR” and lowercase versions of all drug names. Statistics on the occurrences of the drug names in the custom training dataset can be found in Table 7 in Appendix B. Extracting ADEs and drug names from the same domain ensures a high likelihood of compatibility between ADEs and medications.

The ADE entities extracted from PsyTAR are user-generated descriptions of ADEs that are often multi-word expressions and which use non-standardized terms. We did not correct grammar and spelling errors in the extracted ADEs.

We created the templates in a way that most short noun phrases[6] fit as ADE entities, therefore, short noun phrases were filtered from the ADE mentions in PsyTAR. A total of 1,227 unique ADEs were extracted, amounting to 36.50% of unique ADE entities in PsyTAR.[7]

For the *Positive Sentiment* test, the extracted ADEs were manually filtered to collect 60 less severe ADEs. This was a necessary step to avoid creating unrealistic test cases such as *“I always have severe pain in my hands when I’m on Cymbalta, but I am happy my symptoms have reduced”*.

The time entities for the variations in *Temporal Order* tests were not extracted but generated. Numbers between 1 and 25 inclusive were combined with a noun (either “days”, “weeks”, or “months”). A random selection of these combinations was used as time entities.

5 Results

The following presents the results of both the baselines and the template-based test cases.

5.1 Model Baseline

The results of the baseline models can be found in Table 4. All models were evaluated on the same test split of the fine-tuning corpus.

The F_1-score of BioRedditBERT on the positive class (ADE) is 0.698, whereas XLM-RoBERTa achieves a score of 0.700, which indicates very similar general performance. Due to the large class imbalance, the models reached a higher performance on the majority class (no ADE) with F_1-scores of 0.972. The high overlap in data allows for comparison of this model’s performance to the best performing models proposed in the latest SMM4H Shared Task on ADE classification (Weissenbacher et al., 2022).

Model	**Class**	**P**	**R**	**F$_1$**
BioRedditBERT	ADE	0.720	0.676	**0.698**
	no ADE	0.969	0.975	0.972
XLM-RoBERTa	ADE	0.720	0.681	**0.700**
	no ADE	0.970	0.975	0.972

Table 4: The results of the baseline models in precision (P), recall (R), and F_1-score on the test split. Positive class F_1 is highlighted as the most popular metric. All scores are very close which would indicate that we can expect similar task understanding of the models.

5.2 Template-Based Test Results

We compare model performance on the custom dataset to each template-based capability test performance separately. Due to the variations in model performance over the two classes, we use per-class recall as a measurement of comparison between the model performance on the custom dataset and the template-based test cases as shown in Figure 3. For both models, all tests with no ADE labels fall short of the baseline performances. The highest level of performance is observed in the *Negation* tests where BioRedditBERT and XLM-RoBERTa pass 92% and 94% of the test cases, respectively. On the other hand, the *Beneficial Effect* tests perform strikingly worse than the baselines with BioRedditBERT and XLM-RoBERTa passing only 7.5% and 5.8% of the test cases, respectively. All three versions of the negative class *Temporal Order* tests lie below the baselines but to a varying degree with a range of 54%-78% for BioRedditBERT and a range of 62%-74% for XLM-RoBERTa.

For ADE, the models perform below the baseline (recall of 68% for both models) on the *standard Temporal Order* and *double time entities Temporal Order* test (25%-48%), while the baseline is exceeded on the *single time entity Temporal Order* test with 90% for BioRedditBERT and 80% for XLM-RoBERTa. Based on the varying model performance on different types of *Temporal Order* tests for both the negative and the positive class,

[6] The longest extracted ADE has a length of 7 tokens.

[7] More details on the extraction can be found in Appendix A.1.

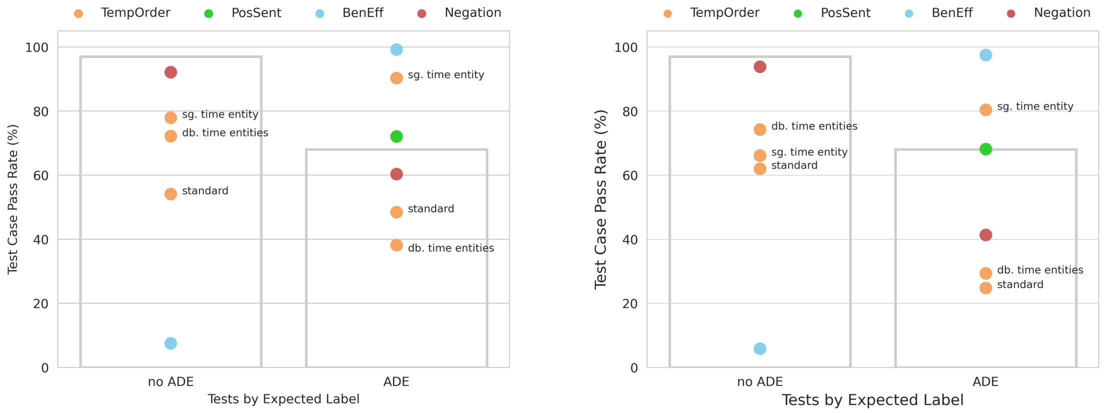


Figure 3: Per-class performance of fine-tuned BioRedditBERT (left) and XLM-RoBERTa (right) on the test set (grey box baseline) and the capability-based test cases. The three distinct types of *Temporal Order* tests refer to variety of *Temporal Order* templates (*standard*, *single* and *double time entity*) highlighted in Table 1. Most test cases are more difficult for the model to solve than the samples from the custom dataset. The biggest difference between the models is the performance on the negation test cases with ADE label, where BioRedditBERT solves 20% more test cases than XLM-RoBERTa. Furthermore, both models have different performances for *Temporal Order* test cases, especially standard cases with ADE label.

one can conclude that the model is not robust to changes in expression of temporal structure: The use of single time entities affects the model performance positively compared to the use of prepositions (*standard Temporal Order*) and double relational time entities. Furthermore, BioRedditBERT (48%) performs much better on *standard Temporal Order* tests than XLM-RoBERTa (25%).

Mild ADEs expressed in positive sentiment as in the *Positive Sentiment* test do not pose a problem to the model. The performance on the *Positive Sentiment* test cases (72% for BioRedditBERT and 68% for XLM-RoBERTa) lies above the baseline of the positive class for both models. Also, the models' performance on the positive class negation test lies below the baseline, with BioRedditBERT (60%) again performing much better than XLM-RoBERTa (41%).

Unlike for the negative class test, almost all test cases in the *Beneficial Effect* test on the positive class are correctly classified as ADE. The poor performance on the negative *Beneficial Effect* test and the outstanding performance on the positive class *Beneficial Effect* test leads to the conclusion that the model has not learned to distinguish between ADEs and beneficial effects. Both models classify 96% of Beneficial Effects test cases as ADE, even though half of the tests have a no ADE mention. Possible explanations for this behavior are that the number of beneficial effect samples in the custom dataset is low and/or that the model does not take the context into account that distinguishes an ADE from a beneficial effect.

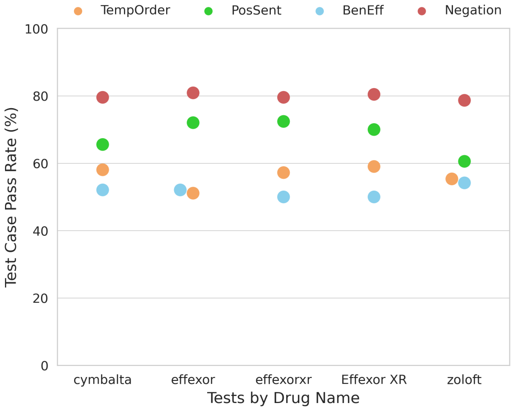


Figure 4: Performance of XLM-Roberta on test cases by drug name and by capability. The number of test cases per capability and drug name is 1440 (Temporal Order), 540 (Positive Sentiment), 48 (Beneficial Effect), 225 (Negation).

Each of the five selected drug name variants was used in every template allowing for an analysis of the impact of drug names in the test cases. Performance variations on test cases with different drug names indicate reduced robustness of the model. We find slight variations in the model performance over different drug names as shown in figure Figure 4 for XLM-Roberta. A potential explanation of these variations may be deviations in the occurrence of the respective drug names in the custom fine-tuning dataset, see Table 7 in Appendix B.

6 Conclusion and Future Work

In this work, we present a template-based approach for evaluating capabilities of models on the task of ADE detection in social media texts. Four capabilities, *Temporal Order*, *Positive Sentiment*, *Beneficial Effect*, and *Negation*, were identified and corresponding tests were created. Two high-performing models for the task of ADE detection were evaluated using the adapted approach.

Results show that the models' performances vary across capabilities. While both models perform well on the *Positive Sentiment* tests, BioRedditBERT outperforms XLM-RoBERTa on *Negation*. The models are not able to distinguish between ADEs and beneficial effects and are not robust to changes in the expression of temporal structure in text. In summary, the template-based approach adapted to ADE classification has provided a better understanding of the shortcomings of high-performing models and can highlight previously undetected differences between models that perform almost identically on a held-out test set. We publish the templates to enable researchers to evaluate their own ADE classification models.

Further research may expand on this work by adding tests for more capabilities and evaluating other models using this approach. For example, in the phenomena annotation described in Section 3.2, we found 1.6% questions and 1.1% speculative content in the tweets. The linguistic variety of the templates could be improved by using a large language model to generate templates or test cases. A different direction of research may focus on improving the model's faults detected during evaluation. One method of improvement is to include a subset of the test cases as new training data (McCoy et al., 2019).

7 Limitations

While the approach of generating new inputs by templates undoubtedly has benefits, it also introduces some limitations. For instance, the combination of all entity fill-ins with all templates can produce some unnatural phrases. An example of this is the *Temporal Order* template "After taking {drug}, I had {ade}.". The ADE entity "weight gain" creates the unnatural sounding test case "After taking cymbalta, I had weight gain." instead of "After taking cymbalta, I gained weight." The unnatural use of language may introduce a bias. This should be kept in mind when using the templates. However, not all entity fill-ins will introduce such a bias and the model's performance on the test cases cannot be fully attributed to the effect of unnatural language use.

A second potential bias when using templates is that it may not be able to depict a large variety of language when only few templates were used. An example of this are the templates for the positive class *Beneficial Effect* test where each test case includes the word "problem". A model could use this as a proxy for correctly classifying the test cases.

Lastly, as described in Section 4, not all features of social media tests were used when creating templates. No anonymized usernames, hashtags, non-standard punctuation, and colloquialisms other than contractions were applied in the templates. This may introduce a bias as there is a slight difference in language variety between the templates and the training data. A researcher should keep in mind that slight changes in the model performance may also be attributed to this shift in language variety.

Acknowledgments

We would like to thank the anonymous reviewers for their feedback on the paper. We further thank our annotators Alon Drobickij, Selin Yeginer, and Emiliano Valdes Menchaca.

This work was partially supported by the German Federal Ministry of Education and Research as part of the project TRAILS. We are furthermore grateful for the support of the German Research Foundation (DFG) under the trilateral ANR-DFG-JST AI Research project KEEPHA (DFG-442445488), and the German Federal Ministry of Education and Research under the grant BIFOLD24B.

References

Hiba Ahsan, Emmie Ohnuki, Avijit Mitra, and Hong You. 2021. Mimic-sbdh: A dataset for social and behavioral determinants of health. In *Machine Learning in Health Care*.

Hassan Alhuzali and Sophia Ananiadou. 2019. Improving classification of adverse drug reactions through using sentiment analysis and transfer learning. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 339–347, Florence, Italy. Association for Computational Linguistics.

Marco Basaldella, Fangyu Liu, Ehsan Shareghi, and Nigel Collier. 2020. COMETA: A corpus for medical entity linking in the social media. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3122–3137, Online. Association for Computational Linguistics.

Yonatan Belinkov and Yonatan Bisk. 2019. Synthetic and natural noise both break neural machine translation. *Conference paper at ICLR 2018*.

Shaily Bhatt, Rahul Jain, Sandipan Dandapat, and Sunayana Sitaram. 2021. A case study of efficacy and challenges in practical human-in-loop evaluation of NLP systems using checklist. In *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*, pages 120–130, Online. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Oscar M. Cumbicus-Pineda, Itziar Gonzalez-Dios, and Aitor Soroa. 2021. Linguistic capabilities for a checklist-based evaluation in automatic text simplification. In *Proceedings of the First Workshop on Current Trends in Text Simplification (CTTS 2021) co-located with the 37th Conference of the Spanish Society for Natural Language Processing (SEPLN2021) Online (initially located in Málaga, Spain)*, pages 70–83.

Rachel Ginn, Pranoti Pimpalkhute, Azadeh Nikfarjam, and Apurv Patki. 2014. Mining Twitter for Adverse Drug Reaction Mentions: A Corpus and Classification Benchmark. In *Proceedings of the Fourth Workshop on Building and Evaluating Resources for Health and Biomedical Text Processing*, pages 1–8.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23.

Harsha Gurulingappa, Abdul Mateen Rajput, Angus Roberts, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. 2012. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *Journal of Biomedical Informatics*, 45(5):885–892.

Andrey Gusev, Anna Kuznetsova, Anna Polyanskaya, and Egor Yatsishin. 2020. BERT implementation for detecting adverse drug effects mentions in Russian. In *Proceedings of the Fifth Social Media Mining for Health Applications Workshop & Shared Task*, pages 46–50, Barcelona, Spain (Online). Association for Computational Linguistics.

Md Mosharaf Hossain, Dhivya Chinnappa, and Eduardo Blanco. 2022. An analysis of negation in natural language understanding corpora. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 716–723, Dublin, Ireland. Association for Computational Linguistics.

Trung Huynh, Yulan He, Alistair Willis, and Stefan Rueger. 2016. Adverse Drug Reaction Classification With Deep Neural Networks. *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics*, pages 877–887.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. 2021. Dynabench: Rethinking benchmarking in NLP. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online. Association for Computational Linguistics.

Robert Leaman, Laura Wojtulewicz, Ryan Sullivan, Annie Skariah, Jian Yang, and Graciela Gonzalez. 2010. Towards Internet-age pharmacovigilance: Extracting adverse drug reactions from user posts in health-related social networks. In *Proceedings of the 2010 Workshop on Biomedical Natural Language Processing*, pages 117–125, Uppsala, Sweden. Association for Computational Linguistics.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv*, abs/1711.05101.

Arjun Magge, Ari Klein, Antonio Miranda-Escalada, Mohammed Ali Al-garadi, Ilseyar Alimova, Zulfat Miftahutdinov, Eulalia Farre-Maduell, Salvador Lima Lopez, Ivan Flores, Karen O'Connor, Davy Weissenbacher, Elena Tutubalina, Abeed Sarker, Juan M Banda, Martin Krallinger, and Graciela Gonzalez-Hernandez, editors. 2021a. *Proceedings of the Sixth Social Media Mining for Health (#SMM4H) Workshop and Shared Task*. Association for Computational Linguistics, Mexico City, Mexico.

Arjun Magge, Elena Tutubalina, Zulfat Miftahutdinov, Ilseyar Alimova, Anne Dirkson, Suzan Verberne, Davy Weissenbacher, and Graciela Gonzalez-Hernandez. 2021b. DeepADEMiner: A deep learning pharmacovigilance pipeline for extraction and normalization of adverse drug event mentions on Twitter. *Journal of the American Medical Informatics Association*, 28(10):2184–2192.

Marta Marchiori Manerba and Sara Tonelli. 2021. Fine-grained fairness analysis of abusive language detection systems with CheckList. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 81–91, Online. Association for Computational Linguistics.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Zulfat Miftahutdinov, Andrey Sakhovskiy, and Elena Tutubalina. 2020. KFU NLP team at SMM4H 2020 tasks: Cross-lingual transfer learning with pretrained language models for drug reactions. In *Proceedings of the Fifth Social Media Mining for Health Applications Workshop & Shared Task*, pages 51–56, Barcelona, Spain (Online). Association for Computational Linguistics.

Milad Moradi and Matthias Samwald. 2022. Improving the robustness and accuracy of biomedical language models through adversarial training. *Journal of Biomedical Informatics*, 132:104114.

Azadeh Nikfarjam and Graciela H. Gonzalez. 2011. Pattern mining for extraction of mentions of adverse drug reactions from user comments. In *AMIA annual symposium proceedings*, volume 2011, page 1019. American Medical Informatics Association.

Azadeh Nikfarjam, Abeed Sarker, Karen O'connor, Rachel Ginn, and Graciela Gonzalez. 2015. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *Journal of the American Medical Informatics Association*, 22(3):671–681.

Charvi Rastogi, Marco Tulio Ribeiro, Nicholas King, Harsha Nori, and Saleema Amershi. 2023. Supporting human-AI collaboration in auditing LLMs with LLMs. In *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '23, pages 913–926. Association for Computing Machinery.

Marco Tulio Ribeiro and Scott Lundberg. 2022. Adaptive testing and debugging of NLP models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3253–3267. Association for Computational Linguistics.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Semantically equivalent adversarial rules for debugging NLP models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 856–865, Melbourne, Australia. Association for Computational Linguistics.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Abeed Sarker, Maksim Belousov, Jasper Friedrichs, Kai Hakala, Svetlana Kiritchenko, Farrokh Mehryary, Sifei Han, Tung Tran, Anthony Rios, Ramakanth Kavuluru, et al. 2018. Data and systems for medication-related text classification and concept normalization from twitter: insights from the social media mining for health (smm4h)-2017 shared task. *Journal of the American Medical Informatics Association*, 25(10):1274–1283.

Simone Scaboro, Beatrice Portelli, Emmanuele Chersoni, Enrico Santus, and Giuseppe Serra. 2021. NADE: A benchmark for robust adverse drug events extraction in face of negations. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, pages 230–237, Online. Association for Computational Linguistics.

Isabel Segura-Bedmar, Ricardo Revert, and Paloma Martínez. 2014. Detecting drugs and adverse events from Spanish social media streams. In *Proceedings of the 5th International Workshop on Health Text Mining and Information Analysis (Louhi)*, pages 106–115, Gothenburg, Sweden. Association for Computational Linguistics.

Thinh Hung Truong, Yulia Otmakhova, Timothy Baldwin, Trevor Cohn, Jey Han Lau, and Karin Verspoor. 2022. Not another negation benchmark: The NaN-NLI test suite for sub-clausal negation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 883–894, Online only. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Davy Weissenbacher, Juan Banda, Vera Davydova, Darryl Estrada Zavala, Luis Gasco Sánchez, Yao Ge, Yuting Guo, Ari Klein, Martin Krallinger, Mathias Leddin, Arjun Magge, Raul Rodriguez-Esteban, Abeed

Sarker, Lucia Schmidt, Elena Tutubalina, and Graciela Gonzalez-Hernandez. 2022. Overview of the seventh social media mining for health applications (#SMM4H) shared tasks at COLING 2022. In *Proceedings of The Seventh Workshop on Social Media Mining for Health Applications, Workshop & Shared Task*, pages 221–241, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Davy Weissenbacher, Abeed Sarker, Arjun Magge, Ashlynn Daughton, Karen O'Connor, Michael J. Paul, and Graciela Gonzalez-Hernandez. 2019. Overview of the Fourth Social Media Mining for Health (SMM4H) Shared Tasks at ACL 2019. In *Proceedings of the Fourth Social Media Mining for Health Applications (#SMM4H) Workshop & Shared Task*, pages 21–30, Florence, Italy. Association for Computational Linguistics.

Yuqing Xie, Yi-An Lai, Yuanjun Xiong, Yi Zhang, and Stefano Soatto. 2021. Regression bugs are in your model! measuring, reducing and analyzing regressions in NLP model updates. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6589–6602, Online. Association for Computational Linguistics.

Maryam Zolnoori, Kin Wah Fung, Timothy B Patrick, Paul Fontelo, Hadi Kharrazi, Anthony Faiola, Yi Shuan Shirley Wu, Christina E Eldredge, Jake Luo, Mike Conway, et al. 2019. A systematic approach for developing a corpus of patient reported adverse drug events: a case study for ssri and snri medications. *Journal of biomedical informatics*, 90:103091.

A Templates

The number of templates with linguistic variations for each capability can be seen in Table 5. Example templates without filled-in entities are in Table 6.

capability	#base templates	#all templates
TempOrder	36	816
PosSent	36	504
BenEff	12	48
Negation	15	137
Total	**99**	**1505**

Table 5: Count of all created templates. Linguistic variation was used to create all templates from base templates.

A.1 Extraction of ADEs from PsyTAR

Sets of Parts of Speech combinations (tagsets) were created to define which sets of POS tags constitute a short noun phrase. An English POS tagger (spaCy) was then used to tag every token in the PsyTAR ADEs and filter out the chosen noun phrases. Examples of PsyTAR ADEs that were retrieved using this method are "listlessness", "recurrence of ocular migraines", and "bad pain in my right arm". The goal of this process was to retrieve as many and diverse ADE descriptions as possible, yet the tagsets are not extensive and not all relevant ADEs were retrieved. Reasons for not passing the tagset filters were not being a noun phrase ("gained 18 pound"), incorrect POS tag assigned tagger ("heartburn"), incorrect POS tags assigned due to typos or extra whitespace, long noun phrases ("stomach cramping the first couple of days"), and punctuation marks/symbols ("increase in alcohol abuse/dependence").

A.2 List of Beneficial Effects

List of (potential) beneficial effects used for the *Beneficial Effect* tests: weight loss/weight gain, sleepiness/decreased need for sleep, loss of appetite/increased appetite.

B Experiment Details

List of entities used as fill-ins for ADE, milder ADE for the *Positive Sentiment* test, and drug names used in the experiments for this project.

- drug names: zoloft, effexor, cymbalta, Effexor XR, effexorxr
- ADEs: Incredible sweet tooth, big appetite, many dreams, Difficulty Orgasming, excellerated heart rate, Insomnia, blackouts, bad pain in my right arm, a little more lethargy, VERY vivid dreams, stiff shoulders, EXTREME DRY MOUTH, Dialated pupils, increase in Libido, acid reflux
- milder ADEs: sugar craving, carbohydrate cravings, bouts of sleeplessness, gum pain, secretion under my toungue, weird dreams, stiff muscles, mild constipation, arm tingling, increased heat sensitivity, strange dreams, poorer concentration, cravings for sweets, hard time falling asleep, neck pain

The counts of the occurrences of the drug names can be found in Table 7.

C Model Details

BioRedditBERT (Basaldella et al., 2020) is a BERT-base uncased model related to BioBERT (Lee et al., 2019), a model pre-trained on the original BERT training corpus (English Wikipedia +

Test Name	Label	Test Description	Example Template
Temporal order standard	no ADE	ADE occurs before drug intake	Before taking {drug}, I experienced {ade}.
	ADE	ADE occurs after drug intake	Before having {ade}, I was put on {drug}.
Temporal order single time entity	no ADE	ADE occurs before drug intake expressed by a time entity	I was experiencing {ade} for {time_entity}, now I started being medicated with {drug}.
	ADE	ADE occurs after drug intake expressed by a time entity	{time_entity} ago I started being treated with {drug}, now I started encountering {ade}.
Temporal order double time entities	no ADE	ADE occurs before drug intake expressed by two related time entities	{time_entity_large} ago I started suffering from {ade}, I have been taking {drug} for {time_entity_small}.
	ADE	ADE occurs after drug intake expressed by two related time entities	I was enduring {ade} for {time_entity_small}, {time_entity_large} ago I started taking {drug}.
Positive Sentiment	ADE	ADE occurrence is reported with positive sentiment	I'm taking {drug} and experiencing {ade}. Still, I am happy my symptoms have reduced.
Beneficial Effect	no ADE	Secondary effect of a drug that is beneficial to the patient	I'm taking {drug} and experiencing weight loss. I'm happy because I was trying to lose weight anyway.
	ADE	Secondary effect of a drug that is an ADE as it is not beneficial	For me, weight loss is a side-effect of {drug}. It's a problem because I am already underweight.
Negation	no ADE	ADE is negated	I am taking {drug} without suffering from {ade}.
	ADE	Statement contains negation, ADE is not negated	That's not true, I took {drug} and encountered {ade}.

Table 6: Overview of all CheckList tests conducted for this project with example templates. Curly brackets in the example templates indicate entity placeholders.

	exact matches	all matches
cymbalta	451	742
effexor	172	312
effexorxr	0	0
Effexor XR	13	23
zoloft	50	100

Table 7: Occurrence of drug names in the fine-tuning training data. Exact matches are case-sensitive. A sample can contain multiple drug name occurrences. ”effexorxr“ was used in the templates without appearing in the training data.

BookCorpus) as well as on medical texts sourced from PubMed and PMC. BioRedditBERT, in turn, was initialized from BioBERT and continued to pre-train on a corpus of health-related Reddit posts. The Reddit dataset contains 800.000 posts from 68 health-related subreddits collected between 2015 and 2018. The specific set of training data of BioRedditBERT was the pivotal argument for choosing this model for the task of ADE classification on the Twitter dataset.

XLM-RoBERTa (Conneau et al., 2020) XLM-RoBERTa is a popular multilingual classification model without a focus on the biomedical domain.

We conducted hyperparameter search for both models and tried batch sizes of 8, 16 and 32 and learning rates of $3 \cdot 10^{-6}$, 10^{-5} and $3 \cdot 10^{-5}$. Both models achieved the best performance on the development set at $16, 10^{-5}$ and trained with the AdamW (Loshchilov and Hutter, 2017) optimizer. No truncation of inputs was applied and the model was evaluated on the validation set after every epoch. The inputs were sampled (batch sampling) with replacement weighted by class ratio due to the class imbalance (see Section 4.1.1).

D Per-Template Performance

The performance of the models on the template-based tests also varies within each test. For all tests except the *Beneficial Effect* tests, the models' performance varies for each template, see Figures 5 and 6. The dependence of the model performance on the template demonstrates that the wording of a template influences the models' ability to handle a capability. In turn, this stresses the importance of creating a wide range of variations in templates when using template-based evaluation.

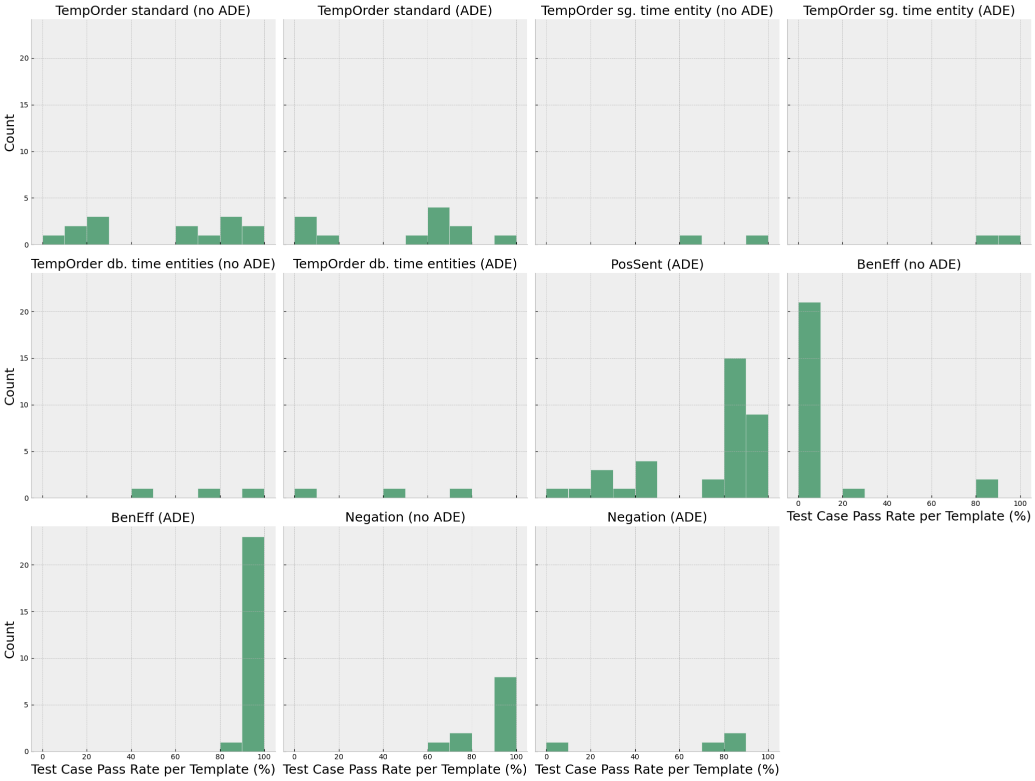


Figure 5: Results of the CheckList tests on the fine-tuned BioRedditBERT by template. The ratio of correctly classified test cases per template is shown on the horizontal axis. Each plot is a histogram showing the count of templates that produced more or less successfully classified test cases.

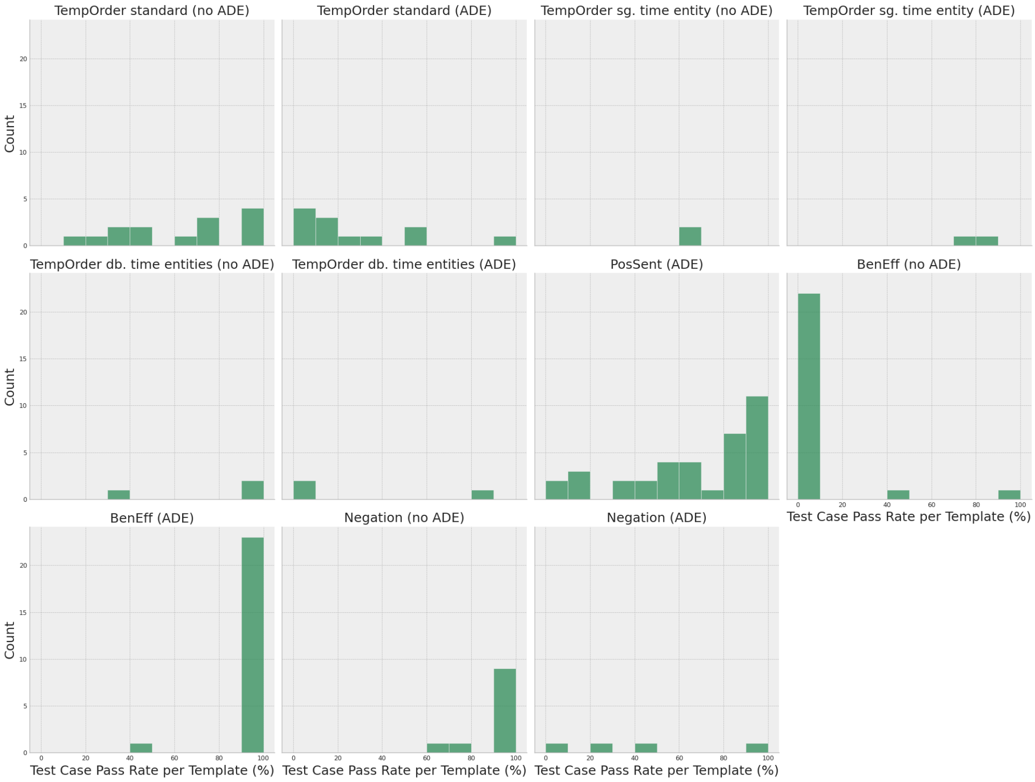


Figure 6: Results of the CheckList tests on the fine-tuned XLM-RoBERTa by template. The ratio of correctly classified test cases per template is shown on the horizontal axis. Each plot is a histogram showing the count of templates that produced more or less successfully classified test cases.

Advancing Healthcare Automation: Multi-Agent System for Medical Necessity Justification

Himanshu Pandey
RISA Labs
himanshu@risalabs.ai

Akhil Amod
RISA Labs
akhil@risalabs.ai

Shivang
RISA Labs
shivang@risalabs.ai

Abstract

Prior Authorization delivers safe, appropriate, and cost-effective care that is medically justified with evidence-based guidelines. However, the process often requires labor-intensive manual comparisons between patient medical records and clinical guidelines, that is both repetitive and time-consuming. Recent developments in Large Language Models (LLMs) have shown potential in addressing complex medical NLP tasks with minimal supervision. This paper explores the application of Multi-Agent System (MAS) that utilize specialized LLM agents to automate Prior Authorization task by breaking them down into simpler and manageable sub-tasks. Our study systematically investigates the effects of various prompting strategies on these agents and benchmarks the performance of different LLMs. We demonstrate that GPT-4 achieves an accuracy of 86.2% in predicting checklist item-level judgments with evidence, and 95.6% in determining overall checklist judgment. Additionally, we explore how these agents can contribute to explainability of steps taken in the process, thereby enhancing trust and transparency in the system.

1 Introduction

In US healthcare, management of administrative workflows represents a pivotal yet formidable challenge. Physicians, nurses, and administrative personnel frequently allocate a substantial portion of their working hours to these procedural tasks, distracting from their primary focus on patient care. One such workflow, Prior authorization (PA) is a healthcare management process used by insurance entities to determine whether a proposed treatment or service is covered under a patient's plan before it is approved to be carried out. This process applies to various treatments and services, including medications, imaging, and procedures (Madhusoodanan et al., 2023). Evaluating a PA application involves assessing medical necessity of patient-specific health records against prevailing coverage guidelines. A major part of these coverage guidelines are clinical guidelines which are systematically developed statements designed to help practitioners make decisions about appropriate health care for specific clinical circumstances. Insurance companies review these clinical guidelines to to justify medical necessity of a procedure or treatment (Chambers et al., 2016).

While Prior Authorization ensures safe, appropriate, cost-effective and evidence based care to all members (Jones et al., 2021), it is a major source of physician and staff burnout as well as job dissatisfaction.There are several ongoing efforts to improve the prior authorization process. High-profile innovations include (1) "gold carding" providers, exempting those who have very high historical approval rates; and (2) automating the process through e-prior auth (e-PA) (Lenert et al., 2023). e-PA proposes a set of transactions conveying the rules for approval in a standardized query representation in CQL. While such rule based methods are adequate for simple authorization decisions, complex cases with temporal data, evidence of responses and trends in clinical data items can be difficult to represent in CQL's rule based format (Lenert et al., 2023). Also, a nationwide survey (Salzbrenner et al., 2022) identified that the use of e-PA was not associated with less provider time or fewer challenges in preparing and submitting PA requests. However, the use of e-PA reported a shorter PA decision time. Additionally, there is an understanding that AI can potentially improve the current state of PA filing (Lenert et al., 2023).

The introduction of Large Language Models (LLMs) (OpenAI, 2024; Touvron et al., 2023) has catalyzed a transformative shift in the capabilities of artificial intelligence, enabling the resolution of complex challenges previously inaccessible to conventional AI methods. LLMs excel in interpret-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 39–49
August 16, 2024. ©2024 Association for Computational Linguistics

ing and synthesizing large volumes of unstructured data, enhancing tasks such as natural language understanding (Yang et al., 2024), sentiment analysis, and automated content creation (Zhou et al., 2024). Building on this foundation, *Multi-Agent Systems*, which employs a collective of AI-powered agents, represents an even further advancement (Guo et al., 2024). This approach decomposes a singular complex task into multiple, manageable sub-tasks and distributes them across multiple agents, each specialized through training for a sub-task. Following this methodology essentially infuses a microservice architecture into the traditional monolithic AI framework, enabling more modular, scalable, and robust AI systems. By integrating the depth and adaptability of LLMs with the collaborative and dynamic nature of Multi-Agent Systems, AI systems can achieve unprecedented levels of performance and versatility across various complex problems (Guo et al., 2024; He et al., 2024).

In this paper, we investigate the application of multi-agent systems for determining medical necessity for a medical procedure. Our contributions are as follows:

- We propose a novel challenge of establishing medical necessity for prior authorizations (PAs) by reasoning on clinical guidelines against patient medical records.
- We decompose the problem statement of PA filing into intermediate sub-tasks, which can then be effectively solved by LLM Agents.
- We demonstrate through extensive experimentations the effect of LLM choice and prompting strategies. Specifically, GPT-4 achieves an accuracy of 86.2% in predicting checklist item-level judgments and 95.6% in determining overall checklist judgment.

2 Related Work

Large Language Models (LLMs) have completely changed the landscape of Natural Language Processing (NLP) in the recent years. LLMs have shown *emergent abilities* (Wei et al., 2022a) in settings like few-shot prompting (Brown et al., 2020) and augmented prompting strategies. Augmented prompting like Chain of Thought (CoT) (Wei et al., 2022b) and Automatic Chain of Thought (Zhang et al., 2022) prompting enables LLMs to solve reasoning tasks using step by step approach. Additionally, instruction fine-tuning with human feedback has made LLMs able to respond to instructions describing unseen tasks (Ouyang et al., 2022). Other advancements include techniques like self-consistency (Wang et al., 2023) which helps LLMs solve complex tasks using multiple different ways of thinking and prompt gradient descent (Pryzant et al., 2023) which edits prompt in the opposite semantic direction of the gradient to boost prompt's performance. Building on this, more dynamic and complex tasks can be tackled by LLM powered Multi Agent Systems (LLM-MAS). These LLM-MAS have collaborative autonomous agents equipped with unique strategies and behaviour (Guo et al., 2024). This agentic behaviour is based on the idea that LLMs can improve in game-play scenario by using previous experiences and feedback (Fu et al., 2023; Madaan et al., 2023).

LLMs have the potential to disrupt medicine. Models like Med-PaLM (Singhal et al., 2022) outperformed state of the art on all MultiMedBench tasks (Tu et al., 2024). GPT-4 has consistently outperformed task-specific fine-tuned models and is comparable to human experts on QA datasets (Zhou et al., 2024). GPT-4 scored 86.65% in United States Medical Licensing Examination (USMLE) where passing percentage was 60% (Nori et al., 2023). It also demonstrates GPT-4's capacity for reasoning about concepts tested in USMLE challenge problems, including explanation, counterfactual reasoning, differential diagnosis, and testing strategies. Some recent researches have started to explore the impact of LLMs in discharge summary generation (Ellershaw et al., 2024; Williams et al., 2024), care planning (Nashwan and Hani, 2023; Jung et al., 2024), Electronic Health Records (EHRs) (Cui et al., 2024; Ahsan et al., 2023). Text-to-SQL parsing has attracted significant interest (Li et al., 2024). Building on this idea, numerous research efforts, such as EHRSQL (Lee et al., 2022), are focused on extracting data from EHRs. Additionally, there are ongoing efforts to develop solutions for EHR-based question-answering tasks (Shi et al., 2024).

However, the domain of PA filing is largely untouched by LLMs mainly because of lack of publicly available data despite the understanding that AI can potentially improve its current state (Lenert et al., 2023). While some efforts have been made to automate PA filing, for example (Diane et al., 2023) where ChatGPT is utilised to generate PA letters for Orthopedic Surgery Practice, but the process lacks the important step of establishing medical ne-

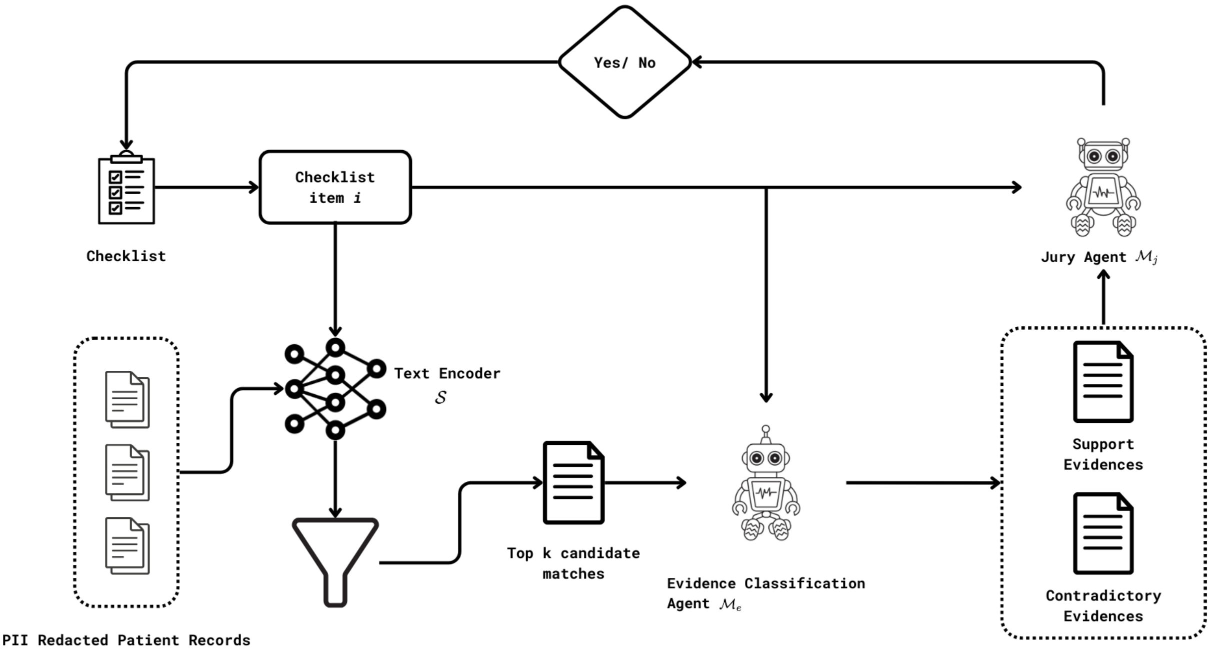

Yes/ No
Checklist
Checklist item i
Jury Agent $\mathcal{M}_j$
Text Encoder $\mathcal{S}$
Support Evidences
Contradictory Evidences
Top k candidate matches
Evidence Classification Agent $\mathcal{M}_e$
PII Redacted Patient Records

Figure 1: Leaf-Level Judgement Prediction where the first agent classifies the documents into supporting and contradictory sets and then the jury agent determines if the checklist item is satisfied.

cessity using AI. Another study aims to determine PA Approval for Lumbar Stenosis Surgery with Machine Learning (De Barros et al., 2023) but it uses surgery specific symptoms as input variables which would be difficult to generalize.

3 Problem Statement

As mentioned above, the evaluation of medical necessity is conducted through a meticulous comparison between patient medical records and established clinical guidelines. These medical records are systematically structured in a json-like format, usually in FHIR [1], within Electronic Health Records (EHRs) systems. Each object (resource) can be of type Patient (Patient Demographics), Observation (Laboratory Results), Procedure (Treatment History), Medication Request, Diagnostic Report etc. We define a set of EHR documents (resources) as $\mathcal{D} = \{d\}_{i=1}^{N_D}$ of size N_D

Further, clinical guidelines are formatted in a hierarchical, tree-like structure (referred as *checklist* in this paper), where each guideline statement (*parent node*) can encompass an arbitrary number of subordinate child statements (*sub-checklist or leaf node*) nested within it as shown in Figure 2 and 3. Thus, we define a coverage guideline or checklist as $\mathcal{C} = \{c\}_{j=1}^{N_C}$, where c is a checklist item.

Eventually, the task is to automatically determine the medical necessity $Y \in \{-1, 0, 1\}$ where -1 means the medical necessity is not justified, 1 means it is justified and 0 means there is a lack of sufficient evidence to justify the medical necessity criteria.

Recognizing the importance of transparency in the task, we also aim to provide evidence $\mathcal{E}_c = \{e_{c_k}\}_{k=1}^{N_c}$. These evidences can be used downstream to cross-reference medical documents used to establish medical necessity for the procedure.

We aim to construct a machine learning model $\mathcal{M}$ such that:

$$\mathcal{M}(\mathcal{D}, \mathcal{C}) = \{Y, \{\mathcal{E}_c\}\} \; \forall c \in \mathcal{C} \qquad (1)$$

4 Methodology

Recently Large Language Models have shown great performance improvements by breaking down complex tasks into simpler sub-problems (Khot et al., 2022). Motivated by this observation, we propose a two step solution for our problem statement. First we determine the judgement of each of the leaf node checklist item. Subsequently, we propagate the solution for parent nodes bottom-up based on its child nodes' judgements.

4.1 Leaf-Node Judgement Prediction

Considering the immense volume of documents in Electronic Health Records (EHRs), we propose a Retrieval-Augmented Setup (Gao et al., 2024).

[1] https://www.hl7.org/fhir/

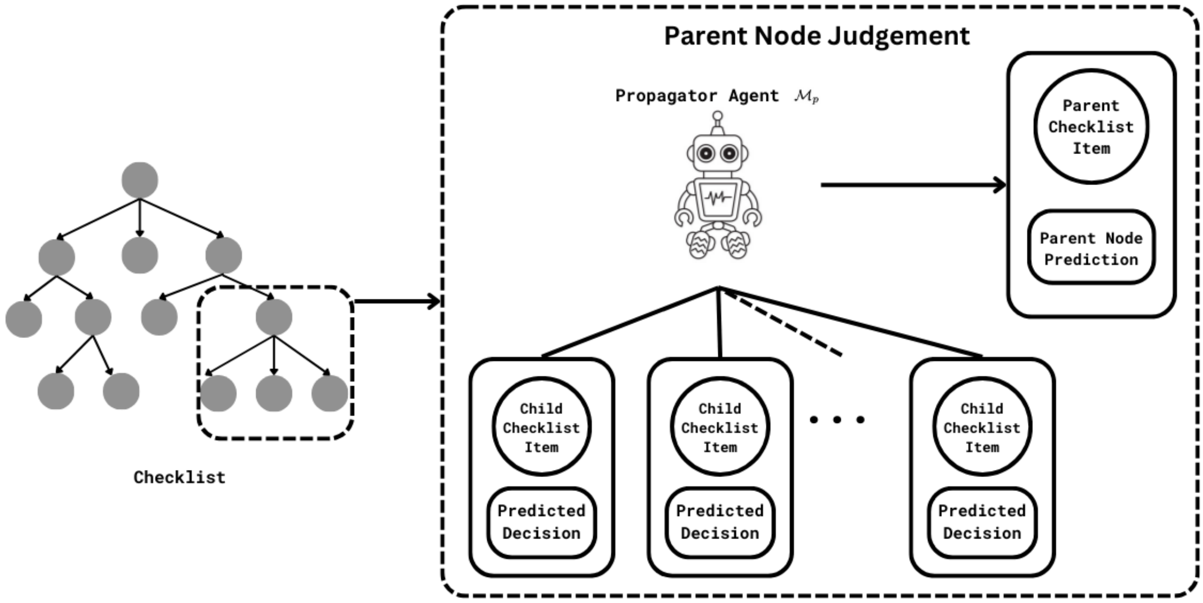


Figure 2: Bottom-Up Judgement Propagation where the agent uses the logical operators contained in a checklist item to determine how the aggregation should take place.

This approach first filters the document pool to identify a set of likely evidences (*top-k evidences*). A *Classification Agent* is then utilized to select the relevant evidences for the specific checklist item, enabling precise and efficient data extraction.

Top-k Evidence Selection: Given the EHR data $\mathcal{D}$, we first decompose it into its constituent resources (documents) where each document is an individual entry (individual lab-report data, procedure etc.) . In order to filter-down documents that are redundant towards the judgment, we first obtain top k candidate matches for the checklist item c from D. To achieve this, we propose to use a text encoder $\mathcal{S}$ to derive semantic representations for each checklist item c and for each document d in the EHR data. This method allows us to map both the checklist items and the documents into a shared semantic space, facilitating more effective matching based on relevance. Subsequently, we employ a semantic similarity metric to calculate the similarity score between each document d and the checklist item c. Based on the similarity metric, we obtain top-k closest matched documents with the checklist item c. Note that due to cost involved in using LLMs for this task, we keep[2] $k < 50$.

$$\mathcal{S}(\{d_i\}|_{i=1}^{N_D}, c) = \{d_{i'}\}|_{i'=1}^{k} \; \forall c \in \mathcal{C} \qquad (2)$$

Evidence Retrieval and Prediction: Our proposed *Evidence Classification Agent* $\mathcal{M}_e$, first looks at each document d_i in top-k evidences retrieved along with the checklist item c and gives a verdict v_i, whether the document d_i is a supporting evidence, a contradictory evidence or it does not affect the judgment y_c. Note that this agent is executed k times since there are k retrieved documents.

$$\mathcal{M}_e(\{d_i, c\}|_{i=1}^{k}) = \{v_i\}|_{i=1}^{k} \; \forall c \in \mathcal{C} \qquad (3)$$

Then our *Jury Agent* $\mathcal{M}_j$ picks up the complete set of evidences $d_i|_{i=1}^{k}$ along with their verdicts $v_i|_{i=1}^{k}$ and predicts the leaf-level checklist item judgment y_c along with evidences $\mathcal{E}_c \subset s_i|_{i=1}^{k}$ that acted in favour of the judgement y_c. We run this leaf-node pipeline multiple times ($n = 10$) and take vote of all predictions to determine the final judgement y_c. Confidence score f_c is calculated as the percentage of times the majority answer is predicted by the agent.

$$\mathcal{M}_j(\{d_i, v_i\}|_{i=1}^{k}, \; c) = \{y_c, f_c, \{\mathcal{E}_c\}\} \; \forall c \in \mathcal{C} \qquad (4)$$

4.2 Parent-Node Judgement Prediction

The value of a parent node is contingent upon the values of its nested child nodes. Hence, we determine parent node's value by aggregating children nodes' values which are connected through logical operators (AND, OR, NOT).

[2] In our experiments section, we show how the performance of our approach varies with k

Bottom-Up Judgement Propagation: In order to obtain the decision over the complete checklist $\mathcal{C}$, we propose to use an iterative bottom-up approach. In another words, we start from the leaf nodes and keep obtaining the judgment of their parent nodes. The iterations are terminated when we obtain the judgment and scores of the root node in the checklist $\mathcal{C}$.

Mathematically, at every iteration i, we choose a set of leaf node checklist items $c_j|_{j=1}^{N_{par}}$ having a common parent checklist item c_{par}, having judgements $y_j|_{j=1}^{N_{par}}$ and confidence scores $f_j|_{j=1}^{N_{par}}$. We then calculate the judgement y_{par} and confidence score f_{par} of parent node as:

$$\mathcal{M}_p(c_{par}, \{c_j, y_j, f_j\}|_{j=1}^{N_{par}}) = \{y_{par}, f_{par}\} \quad (5)$$

where $\mathcal{M}_p$ is our *Propagator Agent*.

5 Data Collection and Annotation

Getting live EHR data for the purpose of this evaluation is difficult, costly and full of regulatory requirements. We therefore used de-identified discharge summaries from MIMIC-IV-Note (Johnson et al., 2023a) as a proxy for this data. All discharge summaries therein have sections like chief complaint, history of present illness, past medical history, social history, physical and lab examinations, medications etc. which serves as the ideal data for this experiment. An average discharge summary has approximately 300 sentences divided into different categories. Joining this data with MIMIC-IV (Johnson et al., 2023b), we can get the CPT/ICD-10 codes associated with each note. We also collected a set of publicly available clinical guidelines (from CMS etc.) pertaining to Cardiology and Oncology and cross referenced the CPT codes in these guidelines to our notes data, thus creating a dataset of (note, guideline) pairs.

An example checklist [3] is shown in Figure 3. The checklist shows the clinical guideline for Therapeutic Footwear which consists of two items associated by **AND** operator. Item 2 in itself is a sub-checklist and will be true if any of the sub-checklist item is True as all of them are connected by **OR** operator.

5.1 Leaf Node Data Annotation

We hired 10 individuals with experience between 6-10 years in PA filing/reviewing both on payer and provider side. They were assigned the task of annotating leaf nodes of a checklist as either True, False or No Information. In case of True and False, the annotator has to also highlight statements in the data section as the evidences for that checklist item as shown in Figure 4. Additionally, each (note, guideline) pair was annotated by 3 different annotators and the final verdict was determined by taking the majority vote of all annotators for that checklist item. Following this, we created a dataset of 281 annotated checklists having 4577 leaf checklist items.

Example Checklist

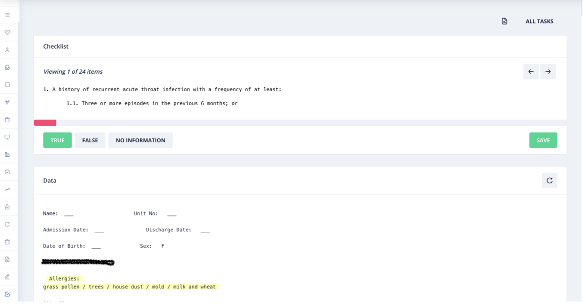
Eligibility Checklist for Therapeutic Footwear

1. The beneficiary has diabetes mellitus; and

2. The certifying physician has documented in the beneficiary's medical record one or more of the following conditions:

   (a) Previous amputation of the other foot, or part of either foot;
   (b) History of previous foot ulceration of either foot;
   (c) History of pre-ulcerative calluses of either foot;
   (d) Peripheral neuropathy with evidence of callus formation of either foot;
   (e) Foot deformity of either foot;
   (f) Poor circulation in either foot;

Figure 3: An example checklist formatted as a decision tree



Checklist

Viewing 1 of 24 items

1. A history of recurrent acute throat infection with a frequency of at least:

   1.1. Three or more episodes in the previous 6 months; or

Data

Name: ___ Unit No: ___

Admission Date: ___ Discharge Date: ___

Date of Birth: ___ Sex: F

[redacted]

Allergies:
grass pollen / trees / house dust / mold / milk and wheat

Figure 4: Annotation Dashboard where each annotator has to mark if the checklist item is True, False or No Information (can't be concluded) and mark evidences for their selection.

5.2 Synthetic Data for Parent Judgement

To test parent-node judgment propagation, we created synthetic data. This was needed because the logical operations required were not within the expertise of our medical domain annotators. To create synthetic data, we first extracted out all sub-checklists from the unique set of guidelines we had,

[3] Taken from CMS: https://www.cms.gov/medicare-coverage-database/search.aspx

and then manually labelled each sub-checklist with the operator (AND, OR and NOT) used for the aggregation of result for that sub-checklist. Then we randomly assigned each leaf node in all sub-checklist their judgements and confidence score and calculated the judgement and confidence score of the parent node programmatically. With different permutations of True, False and No Information used for each sub-checklist, we created a dataset of 4500 sub-checklists used for the evaluation of parent node judgement propagation. This method of synthetic assignment is advantageous as it introduces a range of less likely or extraordinary judgment combinations, thereby challenging our Propagator Agent to maximize its robustness.

6 Experiments and Results

Our experiments were categorized into two distinct segments: assessment of leaf-node judgment and evaluation of parent-node judgment. To facilitate this, we established two separate test environments. Each test-bed was equipped to integrate various Large Language Models (LLMs) to ascertain the optimal model for our needs.

6.1 Leaf-node Judgement

Leaf-node judgment encompassed three sequential tasks. We start by splitting the entire document into sentences. Note that, with MIMIC data it is an easy way to chunk EHR data, but in real case scenario the chunking would happen at FHIR resource [4] level i.e. each Observation, Encounter, Lab Data etc. will act as the smallest chunk that goes into the pipeline. These chunks (or sentences here for simplicity) is first passed through the an encoder module which sorts the sentences according to the cosine similarity. The first 20 sentences are chosen for the experimentation. This simplifies the task of Classification Agent and also saves on LLM cost. The classification agent then segregates these filtered sentences in group of supporting and contradictory evidences which helps predicting the final judgement y_c by the Jury Agent.

Note that the evidences given by the model for each checklist item is not generated but classified. So each evidence will be an exact string match of a sentence from the input document. We have also ensured while annotation that the annotators also selects the evidence from the document as shown in Figure 4. This will help us measure the recall of encoder and classification agent against annotated data. The recall metric is defined as:

$$Recall = \frac{|t_{human} \cap t_{agent}|}{|t_{human}|} \quad (6)$$

where $|t_{human} \cap t_{agent}|$ represents the number of tokens that intersect between the human annotator and the agent, and $|t_{human}|$ is the total number of tokens identified as evidence by the human annotator. This measures if the Jury Agent had enough information to conclude the judgement.

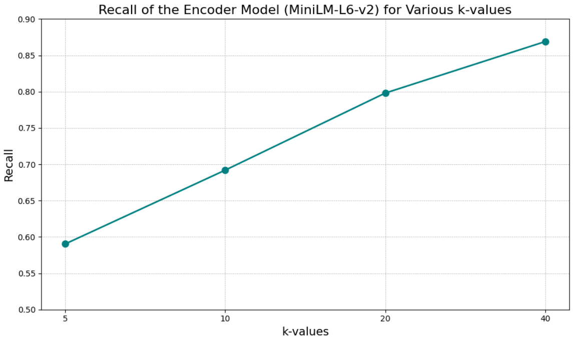


Figure 5: Recall of Encoder (MiniLM-L6-v2) model for various k-values

For encoder model, we used `MiniLM-L6` [5]. We took the top similarity sentences given by encoder model for various k-values and calculated recall against the human evidences and computed the average recall for all checklist items. The results are plotted in Figure 5. For $k = 40$, we get recall as 0.8689, which concludes that using encoder preserves useful information while discarding around 85% of irrelevant data (average MIMIC data has 300 sentences) towards the judgement.

Table 1: Recall metric for Classification Agent with different k values

Model	$\boldsymbol{k = 10}$	$\boldsymbol{k = 20}$
GPT-4	0.5792	0.6741
GPT-3.5	0.4844	0.5554
Claude-Opus	0.5254	0.5845
Calude-Sonnet	0.5042	0.5430

On similar lines, we calculated the recall of the Classification Agent by comparing segregated evidences: if humans marked a checklist item as true, we compared the supporting evidences from the

[4] Refer: https://www.hl7.org/fhir/

[5] Refer: https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

agent to those identified by humans, and similarly, if marked as false, we compared the contradictory evidences. Table 1 shows the recall of Classification Agent for various LLMs. Clearly for $k = 20$ we have significantly higher recall as more evidences were present for the classifier to act upon. GPT-4 outperfomed other models with a maximum recall of 0.67 while other models showed slightly lower values.

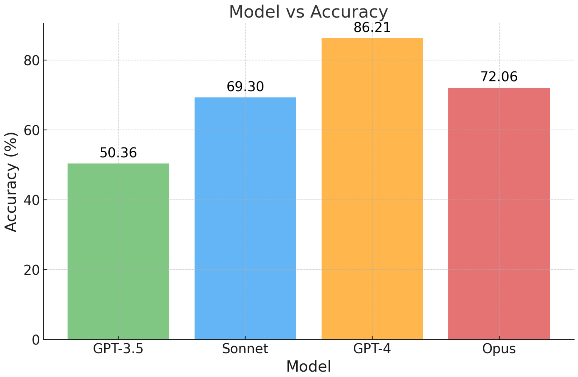


Figure 6: Accuracy of various LLMs for Jury Agent

We conducted a comprehensive evaluation of the Jury Agent ($\mathcal{M}_j$) employing various Large Language Models (LLMs), with a primary focus on accuracy and how it is affected with the number of retrieved evidences (k). GPT-4 and Opus demonstrated robust performance, achieving accuracies of 86% and 72% (Figure 6), respectively. Notably, while Sonnet exhibited a slightly lower accuracy of 69% compared to Opus, it provided a considerable advantage in terms of latency, reducing it by approximately 32% (Figure 7).

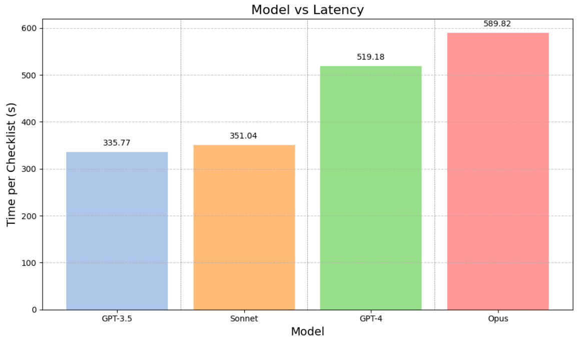


Figure 7: Latency of various LLM model for the leaf-node pipeline

Effect of Number of Retrieved Evidences (*k*): To better understand the effect of *k* on our pipeline and choose the best value we tweaked the value of *k* and ran the pipeline on a smaller sample of our dataset consisting of 20 checklists (having 680 checklist items). We observed that as we increase the value of *k*, the model performance increases till a value of $k = 20$, after which the accuracy gets saturated as shown in Figure 8.

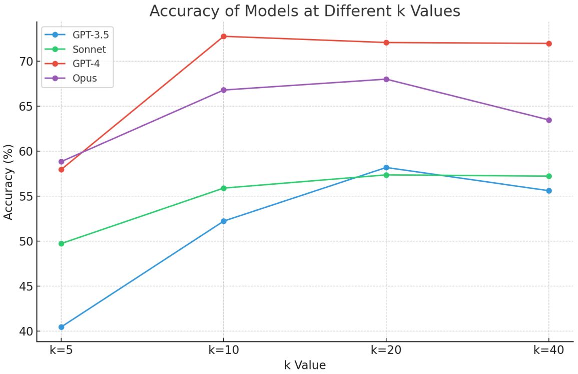


Figure 8: Effect of various k-values on Jury Agent

6.2 Parent-node Judgement

Our *Propagator Agent* is an LLM-powered Agent, which takes up a parent node and its corresponding leaf nodes (along with their judgments and confidence scores) to obtain the judgment and confidence score of the parent node. This was done in two ways. In the first experiment, the LLM agent is asked directly to determine the response and score given parent statement and its child statements, responses and scores. The agent has to understand the logical operators (AND, OR, NOT) and then combine the child responses (True, False, No Information) to conclude parent judgement. The logical rules for No Information items is given in Figure 9 and rules for calculating confidence score is given in Figure 10. In the second experiment, the LLM agent was asked to compute the logical operator between each child item and then the calculation of response and confidence score was done programmatically.

We evaluate the performance of the *Propagator Agent* across various dimensions. The outcomes of this analysis are presented in Table 2. The score accuracy refers to the accuracy of both the response and confidence score propagated correctly while the response accuracy is accuracy of only response being propagated correctly to the parent node resulting from the first experiment. The operator accuracy refers to the accuracy of the model to correctly identify the operators as done in the second experiment.

From the table we can conclude that the Agent

Table 2: Model performance for Propagator Agent using Chain of Thought (CoT) & In-Context Learning (ICL)

	GPT-4		GPT-3.5		Claude-Sonnet		Claude-Opus	
CoT + ICL	**ICL**	**CoT + ICL**	**ICL**	**CoT + ICL**	**ICL**	**CoT + ICL**	**ICL**	**CoT + ICL**
Response Accuracy (%)	87.17	95.60	78.75	91.20	82.05	85.71	85.34	95.24
Score Accuracy (%)	78.35	93.04	48.35	85.34	53.66	81.31	79.12	94.50
Operator Accuracy (%)	89.27	95.01	81.04	92.82	82.05	87.54	84.78	94.04

is able to propagate response more accurately than confidence scores, as propagating confidence score is a more complex task than determining the response which involves only logical operations. Second experiment shows that the accuracy of operator determination task is comparable to the response accuracy determined using first approach. Once the operators are determined, response and confidence score are calculated programatically. Since determining operator would be a one time task (to be done while creating guidelines) taking second approach would get us similar accuracy but at significantly lower cost.

Rule Set for No Information Items

Case I: AND Operator

1. True **AND** No Information = No Information
2. False **AND** No Information = False

Case II: OR Operator

1. True **OR** No Information = True
2. False **OR** No Information = No Information

Case III: NOT Operator

1. **NOT** No Information = No Information

Figure 9: Rule Set for No Information Items followed by Propagator Agent for parent node judgement

Effect of Prompting Strategy: We performed two sets of experiments. The first involved providing *In-Context Learning* (ICL) examples (Min et al., 2022) and measuring accuracy. Larger models such as GPT-4 and Opus yielded strong results, whereas smaller models like Sonnet and GPT-3.5 exhibited suboptimal performance when relying solely on ICL prompts. However, in the second experiment, when supplemented with *Chain of Thought* (CoT) prompting (Wei et al., 2022b), the performance of these smaller models markedly improved, demonstrating how the step-by-step reasoning process aids in decomposing the complex task of propagation into manageable segments. However, the use of Chain of Thought (CoT) prompting substantially increases response times for larger models due to its generation of an increased number of tokens compared to ICL-only prompting. In contrast, the enhancements in performance observed with GPT-3.5 are achieved without a marked increase in latency, particularly when compared to larger models such as Opus and GPT-4 under similar conditions.

Confidence Score (f) Calculation

Case I: AND Operator

1. If final response is True:
   f_{par} = min(f of all True child responses)
2. If final response is False:
   f_{par} = max(f of all False child responses)
3. If final response is No Information:
   f_{par} = min(f of all No Information child responses)

Case II: OR Operator

1. If final response is True:
   f_{par} = max(f of all True child responses)
2. If final response is False:
   f_{par} = min(f of all False child responses)
3. If final response is No Information:
   f_{par} = min(f of all No Information child responses)

Figure 10: Confidence Score calculation rules followed by Propagator Agent for parent node judgement

Effect of LLM Choice: We conducted an evaluation of the *Propagator Agent* utilizing various LLMs, with a particular emphasis on metrics such as accuracy and latency. Opus and GPT-4 emerged as the top performers, achieving approximately 94-95% accuracy when CoT prompting was combined with ICL examples.

GPT-3.5 is ranked second in terms of accuracy but presents significant benefits in reduced latency compared to GPT-4 and Opus, as depicted in Figure 11. Additionally, the operational costs associated with GPT-3.5 are substantially lower. Although selecting the optimal model involves a trade-off, GPT-3.5 stands out as the preferred option when considering a balance among cost, latency, and accuracy. Nonetheless, for scenarios where maximum accuracy is crucial, the larger models such as GPT-4 and Opus are more appropriate.

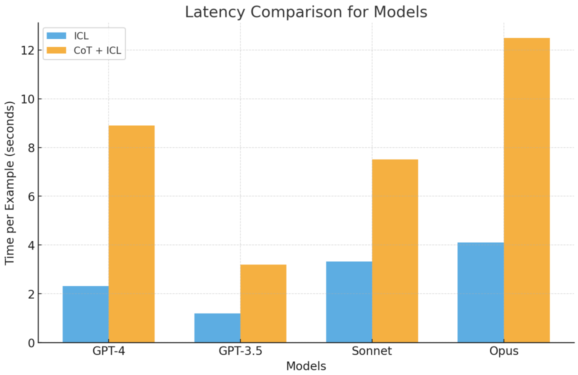


Figure 11: Latency Analysis of LLMs Under ICL and CoT for Propagator Agent when computing score accuracy

7 Conclusion

Our experiments utilized MIMIC-Note data, a set of string-based data. However, real-world applications typically involve obtaining resources (FHIR data) from EHR systems. Converting these resources into stringified data poses a unique engineering challenge. Although manageable, it is crucial to determine whether this data format could impact the effectiveness of our system.

In our approach, we integrated the use of confidence scores. Agents at the leaf nodes compute a confidence score for their predictions, which is then propagated up to the root node alongside the response. The confidence score at the root node is vital as it reflects the system's certainty about the prediction quality. Checklists with low confidence scores are directed to a service layer where experienced professionals can review or adjust the model responses. This feedback loop can be leveraged to refine and enhance future models.

Given our focus on the healthcare sector, ensuring the explainability of outputs from these LLM agents was paramount. The decision-making process was elucidated through Chain of Thought (CoT) prompting and evidence collected by the Classification Agent, enhanced the transparency needed when AI models are employed in healthcare workflows.

While initially designed to automate prior authorization (PA) filing, this solution could also improve clinical decision support (CDS) systems by providing real-time alerts to physicians during consultations. For instance, it could alert physicians to incomplete medical records when prescribing treatments requiring PA, ensuring necessary documentation is promptly addressed. Thus, system responsiveness or latency becomes a critical metric for assessing its performance.

We have shown that breaking down a large, complex problem into smaller, specialized tasks handled by distinct agents can significantly enhance our ability to automate sophisticated tasks that were previously very challenging. This strategy also facilitates the shift from a monolithic AI solution ($\mathcal{M}$) to a micro-service architecture-driven solution ($\mathcal{M}_e$, $\mathcal{M}_j$, and $\mathcal{M}_p$). Currently, our method involves a constrained workflow, but it holds potential for evolving into a system with loosely coupled agents that are more dynamic and capable of improved problem-solving.

The ideal implementation of this methodology would adopt a structure akin to an organization, where the architecture consists of several pods. Each pod contains worker agents specialized in different aspects of the problem, complemented by checker agents that reassess and validate the outputs, triggering reruns when necessary. A super-orchestrator agent would oversee and coordinate the activities across the architecture. This setup aims to mitigate common issues like hallucination often seen in existing LLMs.

8 Acknowledgments

The authors wish to express their gratitude to Gaurav Grontayal and Tanmay Johri for their invaluable assistance in gathering the necessary checklists, hiring consultants and annotators, resolving annotation conflicts, and managing the overall data annotation pipeline. We also extend our thanks to RISA Labs for their support of our research. Special thanks are due to all annotators involved for their diligent work in data annotation.

References

Hiba Ahsan, Denis Jered McInerney, Jisoo Kim, Christopher Potter, Geoffrey Young, Silvio Amir, and Byron C Wallace. 2023. Retrieving evidence from ehrs with llms: Possibilities and challenges. *arXiv preprint arXiv:2309.04550*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

James Chambers, Matthew Chenoweth, and Peter Neumann. 2016. Mapping us commercial payers' coverage policies for medical interventions. *The American journal of managed care*, 22:e323–e328.

Hejie Cui, Xinyu Fang, Ran Xu, Xuan Kan, Joyce C Ho, and Carl Yang. 2024. Multimodal fusion of ehr in structures and semantics: Integrating clinical records and notes with hypergraph and llm. *arXiv preprint arXiv:2403.08818*.

A. De Barros, F. Abel, S. Kolisnyk, et al. 2023. Determining prior authorization approval for lumbar stenosis surgery with machine learning. *Global Spine Journal*, 0(0).

A. Diane, P. Gencarelli, J. M. Lee, et al. 2023. Utilizing chatgpt to streamline the generation of prior authorization letters and enhance clerical workflow in orthopedic surgery practice: A case report. *Cureus*, 15(11):e49680.

Simon Ellershaw, Christopher Tomlinson, Oliver E Burton, Thomas Frost, John Gerrard Hanrahan, Danyal Zaman Khan, Hugo Layard Horsfall, Mollie Little, Evaleen Malgapo, Joachim Starup-Hansen, et al. 2024. Automated generation of hospital discharge summaries using clinical guidelines and large language models. In *AAAI 2024 Spring Symposium on Clinical Foundation Models*.

Yao Fu, Hao Peng, Tushar Khot, and Mirella Lapata. 2023. Improving language model negotiation with self-play and in-context learning from ai feedback. *Preprint*, arXiv:2305.10142.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. 2024. Large language model based multi-agents: A survey of progress and challenges. *Preprint*, arXiv:2402.01680.

Junda He, Christoph Treude, and David Lo. 2024. Llm-based multi-agent systems for software engineering: Vision and the road ahead. *arXiv preprint arXiv:2404.04834*.

Alistair Johnson et al. 2023a. MIMIC-IV-Note: Deidentified Free-Text Clinical Notes (version 2.2). PhysioNet. Accessed: 2023-07-10.

Alistair Johnson et al. 2023b. MIMIC-IV (version 2.2). PhysioNet. Accessed: 2023-07-10.

L.K. Jones, I.G. Ladd, C. Gregor, et al. 2021. Evaluating implementation outcomes (acceptability, adoption, and feasibility) of two initiatives to improve the medication prior authorization process. *BMC Health Services Research*, 21:1259.

HyoJe Jung, Yunha Kim, Heejung Choi, Hyeram Seo, Minkyoung Kim, JiYe Han, Gaeun Kee, Seohyun Park, Soyoung Ko, Byeolhee Kim, et al. 2024. Enhancing clinical efficiency through llm: Discharge note generation for cardiac patients. *arXiv preprint arXiv:2404.05144*.

Tushar Khot, Kyle Richardson, Daniel Khashabi, and Ashish Sabharwal. 2022. Hey ai, can you solve complex tasks by talking to agents? *Preprint*, arXiv:2110.08542.

Gyubok Lee, Hyeonji Hwang, Seongsu Bae, Yeonsu Kwon, Woncheol Shin, Seongjun Yang, Minjoon Seo, Jong-Yeup Kim, and Edward Choi. 2022. Ehrsql: A practical text-to-sql benchmark for electronic health records. In *Advances in Neural Information Processing Systems*, volume 35, pages 15589–15601. Curran Associates, Inc.

Leslie A. Lenert, Steven Lane, and Ramsey Wehbe. 2023. Could an artificial intelligence approach to prior authorization be more human? *Journal of the American Medical Informatics Association*, 30:989–994.

Jinyang Li, Binyuan Hui, Ge Qu, Jiaxi Yang, Binhua Li, Bowen Li, Bailin Wang, Bowen Qin, Ruiying Geng, Nan Huo, et al. 2024. Can llm already serve as a database interface? a big bench for large-scale database grounded text-to-sqls. *Advances in Neural Information Processing Systems*, 36.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. *Preprint*, arXiv:2303.17651.

V Madhusoodanan, L Ramos, IJ Zucker, A Sathe, and R Ramasamy. 2023. Is time spent on prior authorizations associated with approval? *J Nurse Pract*, 19(2):104479. Epub 2022 Nov 10.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *Preprint*, arXiv:2202.12837.

Abdulqadir J Nashwan and Salam Bani Hani. 2023. Enhancing oncology nursing care planning for patients with cancer through harnessing large language models. *Asia-Pacific Journal of Oncology Nursing*, 10(9).

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023. Capabilities of gpt-4 on medical challenge problems. *Preprint*, arXiv:2303.13375.

OpenAI. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.

Reid Pryzant, Dan Iter, Jerry Li, Yin Tat Lee, Chenguang Zhu, and Michael Zeng. 2023. Automatic prompt optimization with "gradient descent" and beam search. *Preprint*, arXiv:2305.03495.

Stephen G Salzbrenner, Carrie McAdam-Marx, Maxwell Lydiatt, Brandon Helding, Lawrence M Scheier, and Patricia Wonch Hill. 2022. Perceptions of prior authorization by use of electronic prior authorization software: A survey of providers in the united states. *Journal of Managed Care & Specialty Pharmacy*, 28(10):1121–1128. PMID: 36125058.

Wenqi Shi, Ran Xu, Yuchen Zhuang, Yue Yu, Jieyu Zhang, Hang Wu, Yuanda Zhu, Joyce C Ho, Carl Yang, and May Dongmei Wang. 2024. Ehragent: Code empowers large language models for few-shot complex tabular reasoning on electronic health records. In *ICLR 2024 Workshop on Large Language Model (LLM) Agents*.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Nathaneal Scharli, Aakanksha Chowdhery, Philip Mansfield, Blaise Aguera y Arcas, Dale Webster, Greg S. Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. 2022. Large language models encode clinical knowledge. *Preprint*, arXiv:2212.13138.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Tao Tu, Shekoofeh Azizi, Danny Driess, Mike Schaekermann, Mohamed Amin, Pi-Chuan Chang, Andrew Carroll, Charles Lau, Ryutaro Tanno, Ira Ktena, Anil Palepu, Basil Mustafa, Aakanksha Chowdhery, Yun Liu, Simon Kornblith, David Fleet, Philip Mansfield, Sushant Prakash, Renee Wong, Sunny Virmani, Christopher Semturs, S. Sara Mahdavi, Bradley Green, Ewa Dominowska, Blaise Aguera y Arcas, Joelle Barral, Dale Webster, Greg S. Corrado, Yossi Matias, Karan Singhal, Pete Florence, Alan Karthikesalingam, and Vivek Natarajan. 2024. Towards generalist biomedical ai. *NEJM AI*, 1(3):AIoa2300138.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. *Preprint*, arXiv:2203.11171.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022a. Emergent abilities of large language models. *Preprint*, arXiv:2206.07682.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022b. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Christopher YK Williams, Jaskaran Bains, Tianyu Tang, Kishan Patel, Alexa N Lucas, Fiona Chen, Brenda Y Miao, Atul J Butte, and Aaron E Kornblith. 2024. Evaluating large language models for drafting emergency department discharge summaries. *medRxiv*, pages 2024–04.

Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Shaochen Zhong, Bing Yin, and Xia Hu. 2024. Harnessing the power of llms in practice: A survey on chatgpt and beyond. *ACM Transactions on Knowledge Discovery from Data*, 18(6):1–32.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022. Automatic chain of thought prompting in large language models. *Preprint*, arXiv:2210.03493.

Hongjian Zhou, Fenglin Liu, Boyang Gu, Xinyu Zou, Jinfa Huang, Jinge Wu, Yiru Li, Sam S. Chen, Peilin Zhou, Junling Liu, Yining Hua, Chengfeng Mao, Chenyu You, Xian Wu, Yefeng Zheng, Lei Clifton, Zheng Li, Jiebo Luo, and David A. Clifton. 2024. A survey of large language models in medicine: Progress, application, and challenge. *Preprint*, arXiv:2311.05112.

Open (Clinical) LLMs are Sensitive to Instruction Phrasings

Alberto Mario Ceballos Arroyo*$^\gamma$ **Monica Munnangi***$^\gamma$ **Jiuding Sun**$^\gamma$
Karen Y.C. Zhang$^\gamma$ **Denis Jered McInerney**$^{\gamma\diamondsuit}$ **Byron C. Wallace**$^\gamma$ **Silvio Amir**$^\gamma$
$^\gamma$Northeastern University $^\diamondsuit$Codametrix
{ceballosarroyo.a, munnangi.m, sun.jiu, zhang.yuchen, b.wallace,s.amir}@northeastern.edu
jmcinerney@codametrix.com

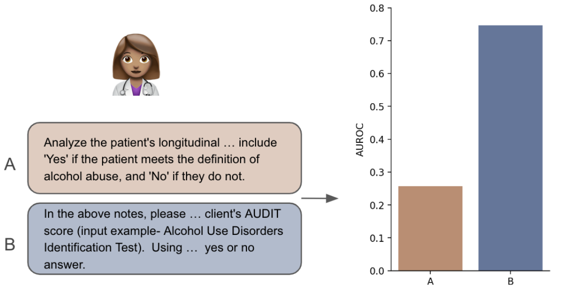
Abstract

Instruction-tuned Large Language Models (LLMs) can perform a wide range of tasks given natural language instructions to do so, but they are sensitive to how such instructions are phrased. This issue is especially concerning in healthcare, as clinicians are unlikely to be experienced prompt engineers and the potential consequences of inaccurate outputs are heightened in this domain.

This raises a practical question: *How robust are instruction-tuned LLMs to natural variations in the instructions provided for clinical NLP tasks?* We collect prompts from medical doctors across a range of tasks and quantify the sensitivity of seven LLMs—some general, others specialized—to natural (i.e., non-adversarial) instruction phrasings. We find that performance varies substantially across all models, and that—perhaps surprisingly—domain-specific models explicitly trained on clinical data are especially brittle, compared to their general domain counterparts. Further, arbitrary phrasing differences can affect fairness, e.g., valid but distinct instructions for mortality prediction yield a range both in overall performance, and in terms of differences between demographic groups.

1 Introduction

Modern LLMs—e.g. GPT-3.5+ (Radford et al., 2019; Ouyang et al., 2022), the FLAN series (Chung et al., 2022), Alpaca (Taori et al., 2023), Mistral (Jiang et al., 2023)—can execute arbitrary tasks *zero-shot*, i.e., provided with only instructions rather than explicit training examples. LLMs have also shown promising improvements in performance on classification and information extraction (IE) tasks, such as named entity recognition (Brown et al., 2020; Munnangi et al., 2024) and relation extraction (Wadhwa et al., 2023a; Ashok and Lipton, 2023; Jiang et al., 2024) in both general and specialized domains like biomedical and scientific literature (Agrawal et al., 2022; Wadhwa et al., 2023b; Asada and Fukuda, 2024).



Analyze the patient's longitudinal … include 'Yes' if the patient meets the definition of alcohol abuse, and 'No' if they do not.

In the above notes, please … client's AUDIT score (input example- Alcohol Use Disorders Identification Test). Using … yes or no answer.

Figure 1: How much does LLM performance on clinical tasks depend on the arbitrary phrasings of instructions? Here we show an illustrative example: Discrepancy in AUROC score for CLINICAL CAMEL on the cohort selection-alcohol abuse classification task, when given the worst (A) and the best (B) performing prompts for ALCOHOL-ABUSE classification task.

However, prior work has shown that LLMs do not "understand" prompts (Webson and Pavlick, 2022) and are sensitive to the particular phrasings of instructions (Lu et al., 2022; Sun et al., 2023). Domain experts in specialized domains such as medicine are especially likely to interact with models by providing instructions (i.e., in *zero-shot* settings), and are unlikely to be talented prompt engineers. For instance, a clinician might task a model to "Extract and summarize the findings of the patient's last X-ray", or ask "When did the patient last receive a painkiller?". It is unrealistic to fine-tune models for every possible such task; hence the appeal of models responsive to arbitrary prompts. A downside, however, is that a clinician's particular phrasing may dramatically affect model performance (Figure 1). Such unpredictability is especially troublesome in healthcare, where poor performance might ultimately impact patient health.

In this work we ask: **How sensitive are LLMs—general and domain-specific—to plausible instruction phrasing variations for clinical tasks?**

[*] Equal contribution

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 50–71
August 16, 2024. ©2024 Association for Computational Linguistics

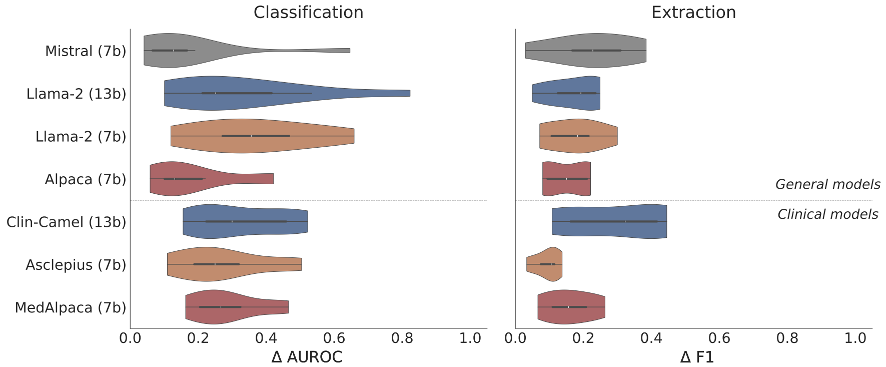


Figure 2: Variance in performance for clinical classification and information extraction tasks for each model. We show the distribution of **deltas between the best and worst performing prompt** for each task.

Our analysis deepens prior work on robustness by focusing on the clinical domain; this is important both due to the higher stakes and because clinical notes differ qualitatively from general domain text. For example, notes in EHR often contain grammatical errors ("*Pt complains of headache, and feel dizzy.*"); abbreviations not defined in context ("*Pt*" could be "*patient*" or "*Prothrombin time*"), and; domain-specific jargon ("*edema*", "*Diuretic*").

Therefore, one of the key aspects we consider is the domain-specificity of models. Are clinical LLMs more (or less) robust to different valid instruction phrasings written by doctors, compared to their general domain counterparts? To assess this, we evaluate recently released LLM variants trained on synthetic datasets comprising automatically generated clinical notes (Kweon et al., 2023), and medical dialogue from case reports found in biomedical literature (Toma et al., 2023). We find that performance varies substantially given alternative instruction phrasings for both general and clinical LLMs. Figure 2 shows the distribution of deltas between the best and worst performing prompts across a set of clinical classification and information extraction tasks.

Finally, we investigate how instruction phrasings impact the fairness of predictions, by which here we mean observed differences in performance between demographic subgroups. The degree to which LLMs might perpetuate and exaggerate such disparities in clinical use is a topic of active research (Omiye et al., 2023; Pal et al., 2023; Zack et al., 2024). Here we contribute to this by investigating the interaction between prompt phrasings and fairness. We find significant performance differences (up to 0.35 absolute difference in AUROC) in a mortality prediction task from MIMIC-III between *White* and *Non-White* subgroups and also a significant disparity between *Male* and *Female* patients (up to 0.19 absolute difference in AUROC). To facilitate future research in this direction, we release our code and prompts[1].

2 Experimental Framework

Our experimental setup is intended to quantify the robustness of LLMs to natural variations in instructional phrasings for clinical tasks. We considered a set of ten clinical classification tasks and six information extraction tasks drawn from MIMIC-III (Johnson et al., 2016) and prior i2b2 and n2c2 challenges,[2] summarized in Table 1 (§2.1). We recruited a diverse group of medical professionals to write prompts for each task (§2.2). We then evaluated the performance, variance, and fairness of seven LLMs (four general-domain and three domain-specific) across prompts (§2.3).

2.1 Tasks and Datasets

MIMIC-III (Johnson et al., 2016) is a database of de-identified EHR comprising over 40k patients admitted to the intensive care unit of the Beth Israel Deaconess Medical Center between 2001 and 2012. It comprises structured variables and clinical notes (e.g., doctor and nursing notes, radiology reports, discharge summaries); we focus on the latter. MIMIC-III also contains demographic information, including ethnicity/race, sex, spoken language, religion, and insurance status (Chen et al., 2019). As an illustrative predictive task, we consider in-hospital mortality prediction, which has been the subject of prior work (Harutyunyan et al., 2017). Owing to compute constraints, we sub-sampled the test-split to 10% of the data (preserving class ratio), yielding 160 records for evaluation.

[1] https://github.com/alceballosa/clin-robust
[2] https://n2c2.dbmi.hms.harvard.edu/

Dataset	Task	Test Set	Task type
MIMIC-III	In-hospital Mortality	160	Binary Classification
Obesity co-morbidity	Asthma	507	Binary Classification
	CAD	507	Binary Classification
	Diabetes	507	Binary Classification
	Obesity	507	Binary Classification
Cohort Selection	Abdominal	86	Binary Classification
	Alcohol-Abuse	86	Binary Classification
	Drug-Abuse	86	Binary Classification
	English	86	Binary Classification
	Decisions	86	Binary Classification
Medical Challenge	Medication	251	Extraction
Relation Challenge	Concept Problem	256	Extraction
	Concept Test	256	Extraction
	Concept Treatment	256	Extraction
Adverse Drug Effects	Drug	202	Extraction
Risk Assessment	Risk Factor CAD	514	Extraction

Table 1: Tasks and datasets used for evaluation.

n2c2 2018 Cohort Selection Challenge (Stubbs and Uzuner, 2019) aims to identify whether a patient meets the criteria for inclusion in a clinical trial based on their longitudinal records. The dataset contains 288 patients, their associated clinical notes and a set of binary labels indicating whether they meet the criteria for each of 13 possible cohorts (e.g., drug abuse, alcohol abuse, ability to make decisions, among others). In this study, we focus on the 5 cohorts shown in Table 1 and treat each as an independent binary classification task aiming to predict whether the criteria is "met" or "not met".

i2b2 2008 Obesity Challenge (Uzuner, 2009) entails identifying patients suffering from obesity and its co-morbidities from their discharge summary notes. The dataset comprises 1027 pairs of de-identified discharge summaries and 16 disease labels from intuitive judgements which are based on the entire discharge summary. We report the performance for obesity and three co-morbidities (i.e., asthma, atherosclerotic cardiovascular disease (CAD), and diabetes mellitus (DM)), each framed as a binary classification task aiming to predict whether the condition is "present" or "absent".

n2c2 2018 Adverse Drug Events and Medication Extraction in EHRs (Henry et al., 2020) consists of a relation extraction task focused on identifying drugs/medications and their relations to adverse events for the patient. The dataset contains 202 patients and we focus only on the named entity recognition portion of the task (i.e. recognizing spans referring to drugs/medications).

i2b2 2014 Identifying Risk Factors for Heart Disease over Time (Stubbs et al., 2015): entails identifying medical risk factors linked to Coronary Artery Disease (CAD) in the EHR of patients with diabetes. The target factors include hypertension, obesity, smoking status, diabetes, hyperlipidemia, family history, and CAD itself. Here we consider only the latter.

i2b2 2010 Relations Challenge (Uzuner et al., 2011) consists of three related tasks: (1) identification of medical problems, tests, and treatments; (2) classification of assertions made on medical problems; and (3) relation extraction concerning medical problems, tests, and treatments. The data for this challenge includes discharge summaries from Partners HealthCare, and the Beth Israel Deaconess Medical Center (Lee et al., 2011), as well as discharge summaries and progress notes from the University of Pittsburgh Medical Center. We conduct evaluation on the first task (i.e. extraction of problems, tests, and treatments) over the notes of 256 patients.

i2b2 2009 Medication Extraction Challenge (Patrick and Li, 2010) focuses on the extraction of medications from clinical notes in the EHR, as well as their modes, reasons and frequency of administration. We center our analysis on medication extraction only, which encompasses around 1250 unique medications over 251 notes.

2.2 Instruction Collection

We hired twenty medical professionals from different professional and demographic backgrounds, with varying medical specialties and years of experience. These included medical doctors (physicians, surgeons), medical writers/editors, nurses, and medical consultants from various countries, such as the United States, Nigeria, Kenya, Canada, Zambia, Egypt, Malawi, Pakistan, Philippines, and Ethiopia. All participants were either native-speakers or proficient in English. It should also be noted that participants were not required to have experience with LLMs but the majority of them reported having used these models in the past.

We provided participants with a description of the tasks including the goal, the expected outputs and a (fictitious) example of a clinical note. We then asked them to write instructions (in English) for each task with the only constraint being that they had to ensure the model outputs a valid label (for classification tasks) or a list of items (for extraction tasks). Figure 9 (Appendix A.1) shows an example of the instructions given for a classification task.

Initially, we ran a smaller scale pilot study consisting of one classification and one extraction task, and recruited participants who successfully completed the tasks. The process took around 5 hours on average and we compensated each participant at a rate of $25/hour. We manually reviewed all written instructions and found that some were of poor quality (e.g., did not adhere to the goals of the task, or did not ensure that the model outputs valid responses). In such cases, we removed the author from the study and discarded all of their instructions. We also removed everyone that did not complete all the tasks, resulting in a final collection of instructions from 12 participants. See Appendix A.1 for illustrative examples of the collected instructions[3].

2.3 Models

We measured the performance, variance and fairness of seven general and domain-specific LLMs on each task, using the instructions written by medical professionals. To assess the impact of clinical instruction tuning, we paired all clinical models with their general domain counterparts. We considered three clinical models: ASCLEPIUS (7B) (Kweon et al., 2023), CLINICAL CAMEL (13B) (Toma et al., 2023), and MEDALPACA (7B) (Han et al., 2023); and their corresponding base models, i.e., LLAMA 2 CHAT (7B), LLAMA 2 CHAT (13B) (Touvron et al., 2023), and ALPACA (7B) (Taori et al., 2023), respectively. We also included MISTRAL IT 0.2 (7B) (Jiang et al., 2023) in our experiments due to its high performance in standard benchmarks.

For all models and datasets, we performed zero-shot inference via prompts with a maximum sequence length of 2048 tokens which included the instruction, the input note, and the output tokens (64 for classification, 256 for extraction). Since most clinical notes were too long to process in a single pass, we followed Huang et al. 2020 and split each note into chunks to be processed independently. For binary classification and prediction tasks, we treated the output for a given input note as positive if at least one of the chunks was predicted to be positive, and negative otherwise. For extraction tasks, we combined the outputs from each chunk into a single set of extractions.

Evaluation: Evaluation with generative models is challenging: Models may not respect the desired output format, or may generate responses that are semantically equivalent but lexically different from references (Wadhwa et al., 2023b; Agrawal et al., 2022). We therefore took predictions from the output distribution of the first generated token by selecting the largest magnitude logit from the set of target class tokens. For extraction tasks, we parsed generated outputs and performed exact match comparison with target spans. We report AUROC scores for classification tasks and F1 scores for extraction tasks.

3 Results

We present our main results for Mortality Prediction and Drug Extraction in Figure 3 — results for the other classification and information extraction tasks can be found in Appendix A.2, Figures 12 and 13, respectively. Most models show significant variability in performance for alternative but semantically equivalent instructions in both classification and extraction tasks. To further examine these observed disparities, we plotted the distri-

[3] The full set of instructions is available in our code repository

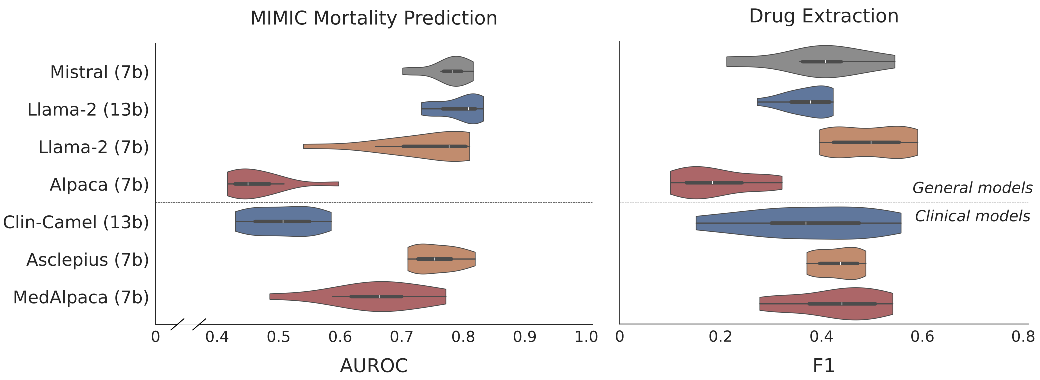


Figure 3: Variability in performance across prompts for the mortality prediction and drug extraction tasks. For most models, different but semantically equivalent prompts yield quite a range of performance.

bution of deltas between the best and worst performing prompts for each task in Figure 2. We see that performance deltas can go up to 0.6 absolute AUROC points for classification tasks and up to 0.4 absolute F1 points for extraction tasks.

In the Mortality Prediction task, we find that LLAMA 2 (13B) outperforms all other models, including the domain-specific ones (Figure 3). However, for the other classification tasks, MISTRAL yields the best results often outperforming the larger models whilst exhibiting less variance (Figure 12). Regarding the clinical models, we observe that ASCLEPIUS consistently attains the best performance in classification tasks albeit with comparable variance.

In the Drug Extraction task, LLAMA 2 (7B) attains the best results on average but with comparable variance to other general LLMs. However, the results for clinical models are mixed: while CLINICAL CAMEL can achieve the highest performance given the best prompt, it also has the highest variance and lowest median performance. MEDALPACA comes close to CLINICAL CAMEL in the best case scenario but with less variance and better median performance. ASCLEPIUS has a median performance similar to that of MEDALPACA but with a much lower variance. We observe similar trends for the other information extraction tasks: LLAMA 2 (7B) consistently outperforms other general LLMs with similar variance, whereas none of the clinical models is clearly superior across tasks — however, ASCLEPIUS seems to have the least variance overall.

To better understand the differences between the general domain and clinical LLMs, we compared their average performance given the best, median and worst prompts. Figures 4 and 5 show the results per model averaged across all classification and extraction tasks, respectively. Surprisingly, we find that general domain models outperform their domain-specific counterparts — with the exception of ALPACA which performs poorly across all tasks. Again we observe that even though CLINICAL CAMEL *can* outperform its general domain analog in extraction tasks given the best prompt, it also shows more variance and much lower performance in the worst case.

Finally, we investigated whether the observed performance variability can be explained by individual differences between experts in prior experience with LLMs or aptitude in writing effective instructions. To assess this, we measured the performance deltas between each prompt and the median prompt for each classification and extraction task. Figure 6 shows the results for LLAMA 2 (7B) and results for other models can be found in Appendix A.2, figures 14 and 15. We find that there are indeed significant differences at the individual level, both in terms of variance and overall performance, particularly for classification tasks. Only roughly half the users can (somewhat) consistently beat the median performance across tasks. We also note these differences can not be solely explained by prior experience with LLMs — some novice users are able to consistently write more effective instructions as compared to other experienced users. However, one caveat is that this prior experience is most likely with larger commercial models which may be more robust to instruction variations.

3.1 Fairness

How do variations in prompt phrasings impact model fairness (here measured as disparities in predictive performance for specific demographic subgroups)? To answer this question, we stratified the patients in the mortality prediction task with

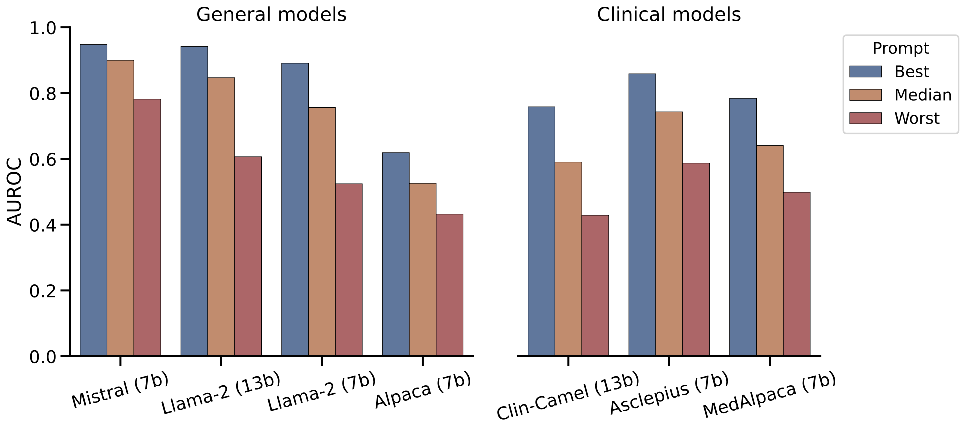


Figure 4: Average AUROC across classification tasks given the best, median, and worst-performing prompts for each model.

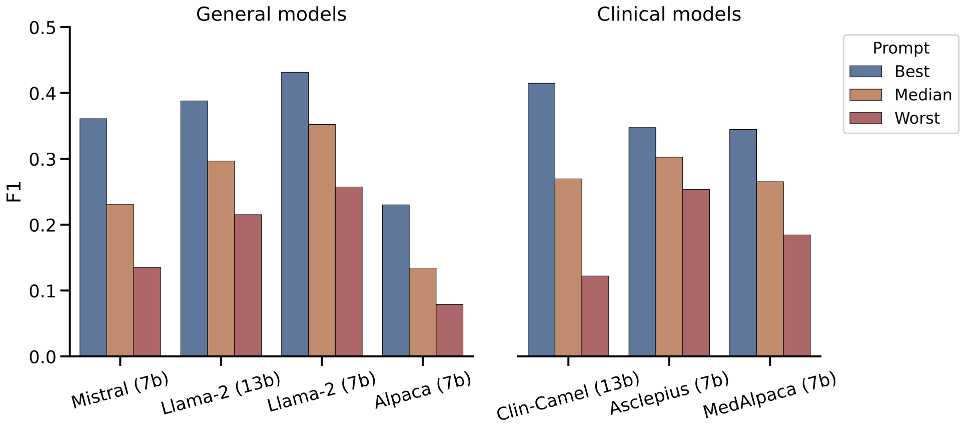


Figure 5: Average F1 across extraction tasks given the best, median, and worst-performing prompts for each model.

		Gender		**Total**
		Female	Male	
Race	White	52	59	111
	Non-White	24	25	49
Total		76	84	160

Table 2: Distribution of gender and race in the sample used examine model fairness (§3.1)

respect to race and sex. To avoid issues with reliability of performance metrics arising from small sub-samples (Amir et al., 2021) we only consider two broad groups (i.e., *White* and *Non-White*). We sorted the instructions according to their overall performance and plot individual subgroup performance (Figure 7). We repeated the analysis for sex (as indicated in EHR) and present individual subgroup performance in Figure 8.

In line with prior work (Amir et al., 2021; Adam et al., 2022), we observe that models have disparate performance for different subgroups. Both LLAMA 2 (7B) and ASCLEPIUS (7B) tend to under-perform for non-White patients compared to White counterparts with absolute differences of up to 0.21 and 0.35 AUROC points, respectively. A possible explanation is that the way in which medical staff write clinical notes differ for White vs Black patients (Adam et al., 2022). However, here non-Whites are an heterogeneous group so there may be other confounding factors.

In regards to sex, we again observe noticeable (albeit smaller) differences in performance with LLAMA 2 (7B) performing worse for *Female* patients across all the prompts with relative differences of up to 0.16 absolute AUROC points, and ASCLEPIUS (7B) yielding differences of up to 0.19 points. Overall, these results indicate that natural variations in prompts may translate to wide differences in fairness. Troublingly, a clinician using such models would likely be unaware that apparently benign phrasing changes may disproportionately affect particular demographic groups.

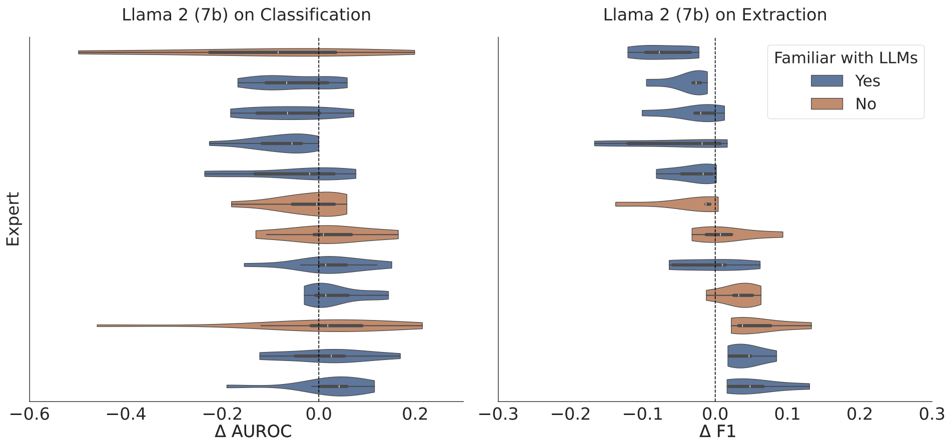


Figure 6: Distribution of performance deltas between each expert's prompt and the median prompt across all tasks. Each violin plot represents an expert color coded according to their familiarity with LLMs.

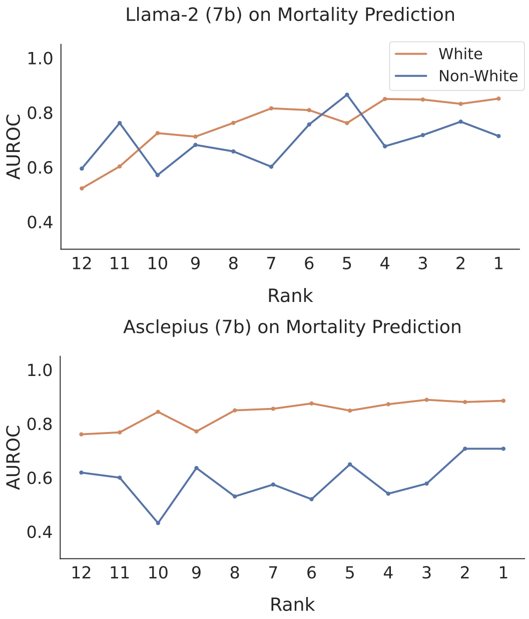


Figure 7: Race subgroup performance on the Mortality Prediction task with a general (top) and clinical model (bottom)

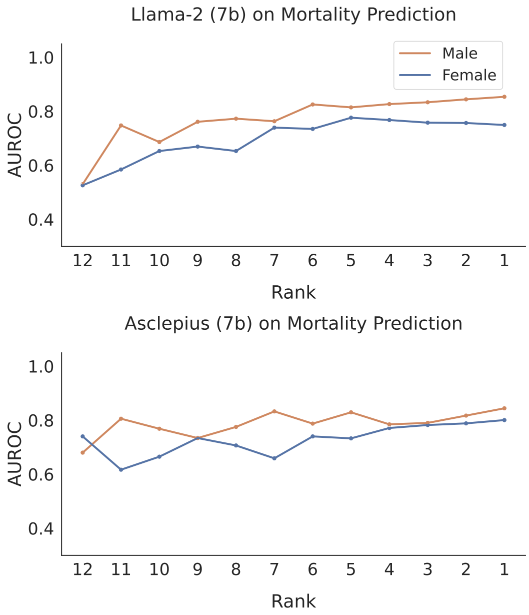


Figure 8: Gender subgroup performance on the Mortality Prediction task with a general (top) and clinical model (bottom)

3.2 Discussion

Our experiments show that instruction-tuned LLMs are not robust to plausible variations in instruction phrasings — equivalent but distinct instructions result in significant differences in both task performance and fairness with respect to demographic subgroups. Moreover, we find that no single model yields optimal performance across tasks, e.g. Mistral 7b is the best model for classification but has middling performance in extraction tasks. We also find that general domain models tend to outperform clinical models — although surprising, these findings corroborate prior work on clinical text summarization (Veen et al., 2023). This may be due to the fact that clinical models are fine-tuned with synthetic or proxy data that does not adequately capture the idiosyncrasies of clinical notes from EHR.

4 Related Work

Instruction-following LLMs Scaling up decoder-only language models imbues them with the ability to solve various tasks given only instructions or a small set of examples at inference time (Brown et al., 2020; Chowdhery et al., 2022). Follow-up work sought to improve this by explicitly training GPT-3 to follow instructions and provide helpful and harmless responses via Reinforcement Learning from Human Feedback (Ouyang et al., 2022; OpenAI, 2022). Others showed that fine-tuning with a causal language modeling objective over labeled data formatted as instruction/response pairs is sufficient to endow even (comparatively) smaller models with instruction-following abilities (Sanh et al., 2021; Wei et al., 2021). This motivated extensive work on compiling large instruction-tuning datasets, such as the Flan 2021 (Chung et al., 2022) and Super-NaturalInstructions collections (Wang et al., 2022), each encompassing over 1600 NLP tasks, and OPT-IML collection with 2000 tasks (Iyer et al., 2022).

LLM Prompt Sensitivity However, LLMs are sensitive to how prompts are constructed (Tjuatja et al., 2023; Raj et al., 2023). In few-shot learning, factors such as the prompt format (Sclar et al., 2023; Chakraborty et al., 2023), as well as the choice (Gutiérrez et al., 2022) and ordering (Lu et al., 2022; Pezeshkpour and Hruschka, 2023) of exemplars have a significant impact on task performance. In zero-shot settings, Webson and Pavlick (2022) found that models often realize similar performance with misleading or irrelevant prompts as with correct ones. Elsewhere, Sun et al. (2023) showed that general domain instruction-tuned LLMs are not robust to variations in instructions — specifically, they found that models underperform when given novel instructions unseen in training. Our work contributes to this line of research by focusing on the clinical domain.

LLMs for Clinical Tasks General domain LLMs encode a surprising amount of clinical and biomedical knowledge allowing them to solve various prediction and information extraction tasks via natural language instructions (Singhal et al., 2023; Agrawal et al., 2022; Munnangi et al., 2024). However, smaller models fine-tuned on task-specific data can outperform generalist LLMs in clinical tasks (Lehman et al., 2023). At the same time, there is a dearth of large high-quality clinical text datasets to train LLMs due to privacy considerations. Researchers have tried to overcome this by exploiting synthetic data generated from biomedical and clinical literature and question answering datasets to train domain-specific models (Toma et al., 2023; Kweon et al., 2023; Han et al., 2023). However, the resulting models are often outperformed by general domain variants (Veen et al., 2023; Excoffier et al., 2024) — our experimental results confirm these observations.

In a contemporaneous study Chang et al. (2024) convened a panel of 80 multidisciplinary experts to red team ChatGPT models for the appropriateness of the responses in medical use cases. Experts were asked to write (non-adversarial) prompts for clinically relevant scenarios and the responses were judged by medical doctors with respect to safety, privacy, hallucinations, and bias. This work is complementary to ours in that it aims to stress test models for the *appropriateness* of their responses to healthcare related prompts whereas we focus on their *sensitivity* to prompt variations.

5 Conclusions

This paper presents a large-scale evaluation of instruction-tuned open-source LLMs for clinical classification and information extraction tasks on clinical notes (from EHR). We specifically focus on model robustness to natural differences in prompts written by medical professionals. We recruited 12 practitioners with different professional and demographic backgrounds, medical specialties, and years of experience to write prompts for 16 clinical tasks spanning binary classification, outcome prediction, and information extraction.

There are a few main generalizable takeaways relevant to machine learning in healthcare in this work. First, the performance LLMs realize on the same clinical task varies substantially across prompts written by different domain experts, and this holds across all models. Second, the domain-specific (clinical) models we evaluated perform, in general, worse than their general domain counterparts. Third, prompt variations have concerning implications for fairness — we find that alternative prompts yield different levels of fairness. Based on these findings, we recommend that practitioners exercise caution when using instruction-tuned LLMs for high stakes clinical tasks which may ultimately impact patient health. Crucially, clinicians using LLMs should be made aware that subtle, plausible variations in phrasings may yield quite different outputs. Beyond healthcare, this work enriches our understanding of (the lack of) LLM robustness and—we hope—will motivate research into new methods to improve models in this respect.

6 Limitations

Our study reveals that open-source instruction-tuned LLMs are sensitive to instruction phrasings and suggests caution in adopting these models for applications that may impact personal health and well-being. However, this work has several limitations. First, we acknowledge that our findings may not generalize to larger commercial models but cost and privacy considerations may preclude the deployment of proprietary models for real-world healthcare applications. Second, we endeavored to recruit a diverse group of medical professionals but our final pool of participants may not be a representative sample of the potential users of these technologies. Moreover, participants were not allowed to see the results of their instructions but in the real world users would have the opportunity to experiment with different prompts and learn how to best use these models. Third, our evaluation protocol for classification tasks may not reflect real world usage — we induced model predictions from the logit distribution of the first generated token. However, in practice users can only see the final generated outputs and must be able to parse and interpret these in the context of the task at hand. Finally, our analysis showed that variations in instructions have implications for fairness with respect to race and gender. However, we did not examine the impact of these disparities on intersectional identities which are often affected by compounded biases.

Acknowledgments

This work was supported in part by National Science Foundation (NSF) award 1901117, and by the National Insitutes of Health (NIH) award R01LM013772.

We also thank the reviewers, for their valuable feedback and comments that helped improve this work.

References

Hammaad Adam, Ming Ying Yang, Kenrick Cato, Ioana Baldini, Charles Senteio, Leo Anthony Celi, Jiaming Zeng, Moninder Singh, and Marzyeh Ghassemi. 2022. Write it like you see it: Detectable differences in clinical notes by race lead to differential model recommendations. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '22. ACM.

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. *Preprint*, arXiv:2205.12689.

Silvio Amir, Jan-Willem van de Meent, and Byron C. Wallace. 2021. On the impact of random seeds on the fairness of clinical classifiers. *Preprint*, arXiv:2104.06338.

Masaki Asada and Ken Fukuda. 2024. Enhancing relation extraction from biomedical texts by large language models. In *International Conference on Human-Computer Interaction*, pages 3–14. Springer.

Dhananjay Ashok and Zachary C. Lipton. 2023. Promptner: Prompting for named entity recognition. *Preprint*, arXiv:2305.15444.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Mohna Chakraborty, Adithya Kulkarni, and Qi Li. 2023. Zero-shot Approach to Overcome Perturbation Sensitivity of Prompts. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5698–5711, Toronto, Canada. Association for Computational Linguistics.

Crystal T. Chang, Hodan Farah, Haiwen Gui, Shawheen Justin Rezaei, Charbel Bou-Khalil, Ye-Jean Park, Akshay Swaminathan, Jesutofunmi A. Omiye, Akaash Kolluri, Akash Chaurasia, Alejandro Lozano, Alice Heiman, Allison Sihan Jia, Amit Kaushal, Angela Jia, Angelica Iacovelli, Archer Yang, Arghavan Salles, Arpita Singhal, Balasubramanian Narasimhan, Benjamin Belai, Benjamin H. Jacobson, Binglan Li, Celeste H. Poe, Chandan Sanghera, Chenming Zheng, Conor Messer, Damien Varid Kettud, Deven Pandya, Dhamanpreet Kaur, Diana Hla, Diba Dindoust, Dominik Moehrle, Duncan Ross, Ellaine Chou, Eric Lin, Fateme Nateghi Haredasht, Ge Cheng, Irena Gao, Jacob Chang, Jake Silberg, Jason A. Fries, Jiapeng Xu, Joe Jamison, John S. Tamaresis, Jonathan H Chen, Joshua Lazaro, Juan M. Banda, Julie J. Lee, Karen Ebert Matthys, Kirsten R. Steffner, Lu Tian, Luca Pegolotti, Malathi Srinivasan, Maniragav Manimaran, Matthew Schwede, Minghe Zhang, Minh Nguyen, Mohsen Fathzadeh, Qian Zhao, Rika Bajra, Rohit Khurana, Ruhana Azam, Rush Bartlett, Sang T. Truong, Scott L. Fleming, Shriti Raj, Solveig Behr, Sonia Onyeka, Sri Muppidi, Tarek Bandali, Tiffany Y. Eulalio, Wenyuan Chen, Xuanyu Zhou, Yanan Ding, Ying Cui, Yuqi Tan, Yutong Liu, Nigam H. Shah, and Roxana Daneshjou. 2024. Red teaming large language models in medicine: Real-world insights on model behavior. *medRxiv*.

Irene Y. Chen, Peter Szolovits, and Marzyeh Ghassemi. 2019. Can AI Help Reduce Disparities in General

Medical and Mental Health Care? *AMA journal of ethics*, 21 2:E167–179.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. PaLM: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Jean-Baptiste Excoffier, Tom Roehr, Alexei Figueroa, Jens-Michalis Papaioannou, Keno Bressem, and Matthieu Ortala. 2024. Generalist embedding models are better at short-context clinical semantic search than specialized embedding models. *Preprint*, arxiv:2401.01943.

Bernal Jiménez Gutiérrez, Nikolas McNeal, Clayton Washington, You Chen, Lang Li, Huan Sun, and Yu Su. 2022. Thinking about GPT-3 In-Context Learning for Biomedical IE? Think Again. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4497–4512.

Tianyu Han, Lisa C Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K Bressem. 2023. MedAlpaca–An Open-Source Collection of Medical Conversational AI Models and Training Data. *arXiv preprint arXiv:2304.08247*.

Hrayr Harutyunyan, Hrant Khachatrian, David C. Kale, and A. G. Galstyan. 2017. Multitask learning and benchmarking with clinical time series data. *Scientific Data*, 6.

Sam Henry, Kevin Buchan, Michele Filannino, Amber Stubbs, and Ozlem Uzuner. 2020. 2018 n2c2 shared task on adverse drug events and medication extraction in electronic health records. *Journal of the American Medical Informatics Association: JAMIA*, 27(1):3–12.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2020. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *Preprint*, arXiv:1904.05342.

Srinivasan Iyer, Xi Victoria Lin, Ramakanth Pasunuru, Todor Mihaylov, Dániel Simig, Ping Yu, Kurt Shuster, Tianlu Wang, Qing Liu, Punit Singh Koura, et al. 2022. OPT-IML: Scaling language model instruction meta learning through the lens of generalization. *arXiv preprint arXiv:2212.12017*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Guochao Jiang, Ziqin Luo, Yuchen Shi, Dixuan Wang, Jiaqing Liang, and Deqing Yang. 2024. Toner: Type-oriented named entity recognition with generative language model. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 16251–16262.

Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li wei H. Lehman, Mengling Feng, Mohammad Mahdi Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3.

Sunjun Kweon, Junu Kim, Jiyoun Kim, Sujeong Im, Eunbyeol Cho, Seongsu Bae, Jungwoo Oh, Gyubok Lee, Jong Hak Moon, Seng Chan You, et al. 2023. Publicly shareable clinical large language model built on synthetic clinical notes. *arXiv preprint arXiv:2309.00237*.

Joon Lee, Daniel J. Scott, Mauricio Villarroel, Gari D. Clifford, Mohammed Saeed, and Roger G. Mark. 2011. Open-access mimic-ii database for intensive care research. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 8315–8318.

Eric Lehman, Evan Hernandez, Diwakar Mahajan, Jonas Wulff, Micah J Smith, Zachary Ziegler, Daniel Nadler, Peter Szolovits, Alistair Johnson, Szolovits Peter Alsentzer, Emily, Alistair Johnson, Emily Alsentzer, et al. 2023. Do we still need clinical language models? In *Conference on Health, Inference, and Learning*, pages 578–597. PMLR.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically Ordered Prompts and Where to Find Them: Overcoming Few-Shot Prompt Order Sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

Monica Munnangi, Sergey Feldman, Byron C Wallace, Silvio Amir, Tom Hope, and Aakanksha Naik. 2024. On-the-fly definition augmentation of llms for biomedical ner. *Preprint*, arXiv:2404.00152.

Jesutofunmi A Omiye, Jenna C Lester, Simon Spichak, Veronica Rotemberg, and Roxana Daneshjou. 2023. Large language models propagate race-based medicine. *NPJ Digital Medicine*, 6(1):195.

OpenAI. 2022. ChatGPT-3.5.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al.

2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Ridam Pal, Hardik Garg, Shashwat Patel, and Tavpritesh Sethi. 2023. Bias amplification in intersectional subpopulations for clinical phenotyping by large language models. *medRxiv*, pages 2023–03.

Jon Patrick and Min Li. 2010. High Accuracy Information Extraction of Medication Information from Clinical Notes: 2009 I2b2 Medication Extraction Challenge. *Journal of the American Medical Informatics Association : JAMIA*, 17(5):524–527.

Pouya Pezeshkpour and Estevam Hruschka. 2023. Large Language Models Sensitivity to The Order of Options in Multiple-Choice Questions. *arXiv preprint*. ArXiv:2308.11483 [cs].

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Harsh Raj, Vipul Gupta, Domenic Rosati, and Subhabrata Majumdar. 2023. Semantic Consistency for Assuring Reliability of Large Language Models. *arXiv preprint*. ArXiv:2308.09138 [cs].

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2023. Quantifying Language Models' Sensitivity to Spurious Features in Prompt Design or: How I learned to start worrying about prompt formatting. *arXiv preprint*. ArXiv:2310.11324 [cs].

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. 2023. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

Amber Stubbs, Christopher Kotfila, Hua Xu, and Ozlem Uzuner. 2015. Identifying risk factors for heart disease over time: Overview of 2014 i2b2/UTHealth shared task Track 2. *Journal of biomedical informatics*, 58(Suppl):S67.

Amber Stubbs and Özlem Uzuner. 2019. New approaches to cohort selection. *Journal of the American Medical Informatics Association*, 26(11):1161–1162.

Jiuding Sun, Chantal Shaib, and Byron C. Wallace. 2023. Evaluating the Zero-shot Robustness of Instruction-tuned Language Models. *arXiv preprint*. ArXiv:2306.11270 [cs].

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford Alpaca: An Instruction-following LLaMA model. https://github.com/tatsu-lab/stanford_alpaca.

Lindia Tjuatja, Valerie Chen, Sherry Tongshuang Wu, Ameet Talwalkar, and Graham Neubig. 2023. Do LLMs exhibit human-like response biases? A case study in survey design. *arXiv preprint*. ArXiv:2311.04076 [cs].

Augustin Toma, Patrick R Lawler, Jimmy Ba, Rahul G Krishnan, Barry B Rubin, and Bo Wang. 2023. Clinical Camel: An Open-Source Expert-Level Medical Language Model with Dialogue-Based Knowledge Encoding. *arXiv preprint arXiv:2305.12031*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Özlem Uzuner. 2009. Recognizing obesity and comorbidities in sparse data. *Journal of the American Medical Informatics Association*, 16(4):561–570.

Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Blüthgen, Anuj Pareek, Malgorzata Polacin, William Collins, Neera Ahuja, Curt P. Langlotz, Jason Hom, Sergios Gatidis, John M. Pauly, and Akshay S. Chaudhari. 2023. Adapted large language models can outperform medical experts in clinical text summarization. *Nature Medicine*, 30:1134–1142.

Somin Wadhwa, Silvio Amir, and Byron Wallace. 2023a. Revisiting relation extraction in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15566–15589, Toronto, Canada. Association for Computational Linguistics.

Somin Wadhwa, Jay DeYoung, Benjamin Nye, Silvio Amir, and Byron C Wallace. 2023b. Jointly extracting interventions, outcomes, and findings from RCT reports with LLMs. In *Machine Learning for Healthcare Conference*, pages 754–771. PMLR.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, et al. 2022. Super-NaturalInstructions: Generalization via Declarative Instructions on 1600+ NLP Tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109.

Albert Webson and Ellie Pavlick. 2022. Do Prompt-Based Models Really Understand the Meaning of Their Prompts? In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2300–2344, Seattle, United States. Association for Computational Linguistics.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Travis Zack, Eric Lehman, Mirac Suzgun, Jorge A Rodriguez, Leo Anthony Celi, Judy Gichoya, Dan Jurafsky, Peter Szolovits, David W Bates, Raja-Elie E Abdulnour, et al. 2024. Assessing the potential of GPT-4 to perpetuate racial and gender biases in health care: a model evaluation study. *The Lancet Digital Health*, 6(1):e12–e22.

A Appendix

A.1 Instruction Collection

To collect instructions from experts, we provided them with a description of the tasks including the goal, the expected outputs and a (fictitious) example of a clinical note. Figure 9 is an example of the instructions given for a classification task; and Figures 10 and 11 show examples of collected instructions. We released the full set of collected instructions along with code.

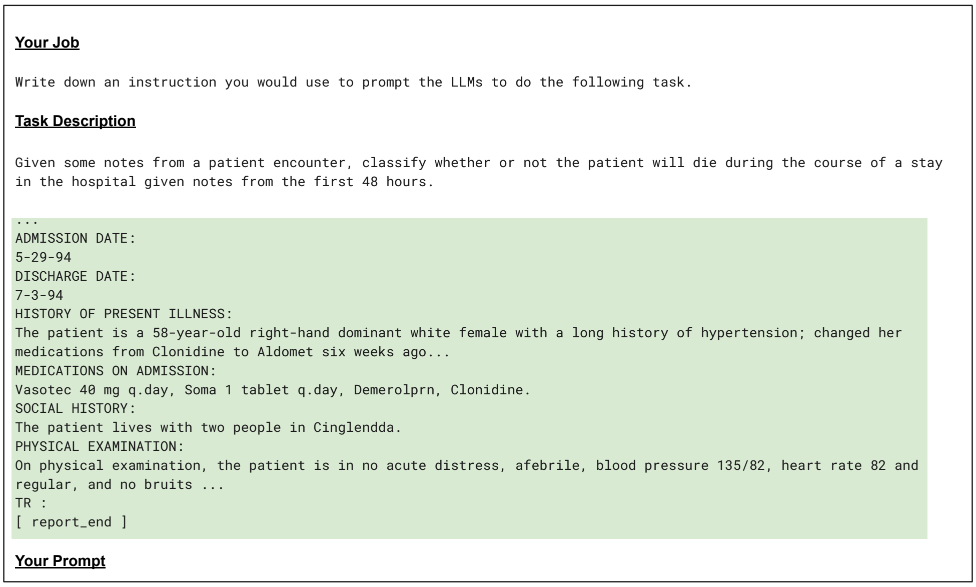
A.2 Results

In this section we present additional results from our experiments. We show detailed results in terms of the mean performance and standard deviation for all the classification and information extraction tasks in tables 3 and 4, respectively.

Figures 12 and 13 plot the variability in performance across classification and extraction tasks, respectively. Figures 14 and 15 plot the deltas in performance between individual expert's prompts and the median prompt per task, for general domain and clinical models, respectively.

Figure 16 show race subgroup performance for the Mortality Prediction task for all the models, and Figure 17 shows a similar analysis for sex.

Our overall results show that, in general, different prompt phrasings yield different performance. Are there prompts that are consistently effective across models? To investigate this, we ranked each prompt with respect to the performance and calculated the median across models. Figures 18 and 19 depict the median performance ranking (among all 12 prompts) achieved by the instructions written by each expert. For classification tasks such as Cohort Abdominal and Cohort Make Decisions, Expert 7 wrote prompts that are consistently among the best performing ones for most models, which is also the case for the prompts written by Expert 11 across five classification tasks. On the other hand, prompts from Expert 2 were consistently among the lower performing ones. A similar pattern can be seen in the extraction tasks, where Experts 6 and 8 wrote some of the best-performing prompts for most of these tasks. This suggests that, to an extent, the performance of prompts is consistent even when tested on different models.



Your Job

Write down an instruction you would use to prompt the LLMs to do the following task.

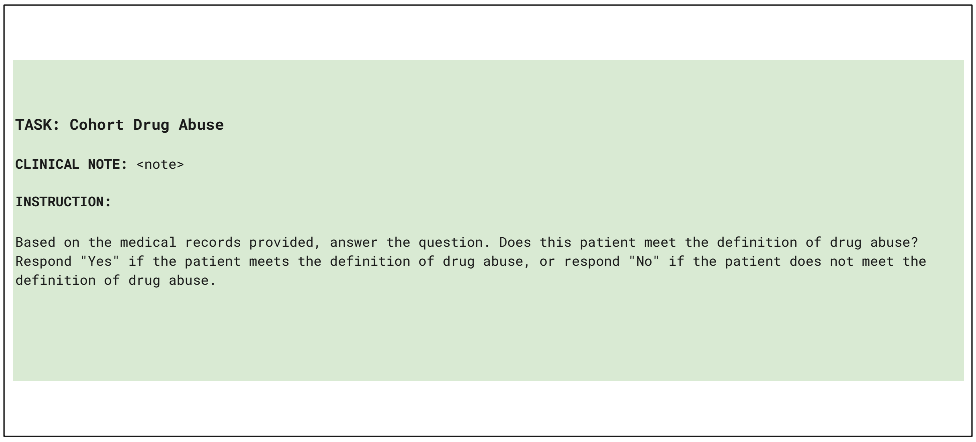
Task Description

Given some notes from a patient encounter, classify whether or not the patient will die during the course of a stay in the hospital given notes from the first 48 hours.

```
...
ADMISSION DATE:
5-29-94
DISCHARGE DATE:
7-3-94
HISTORY OF PRESENT ILLNESS:
The patient is a 58-year-old right-hand dominant white female with a long history of hypertension; changed her
medications from Clonidine to Aldomet six weeks ago...
MEDICATIONS ON ADMISSION:
Vasotec 40 mg q.day, Soma 1 tablet q.day, Demerolprn, Clonidine.
SOCIAL HISTORY:
The patient lives with two people in Cinglendda.
PHYSICAL EXAMINATION:
On physical examination, the patient is in no acute distress, afebrile, blood pressure 135/82, heart rate 82 and
regular, and no bruits ...
TR :
[ report_end ]
```

Your Prompt

Figure 9: Example of instructions for annotators for a classification task; we provided participants with a description of the tasks including the goal, the expected outputs and a (fictitious) example of a clinical note.



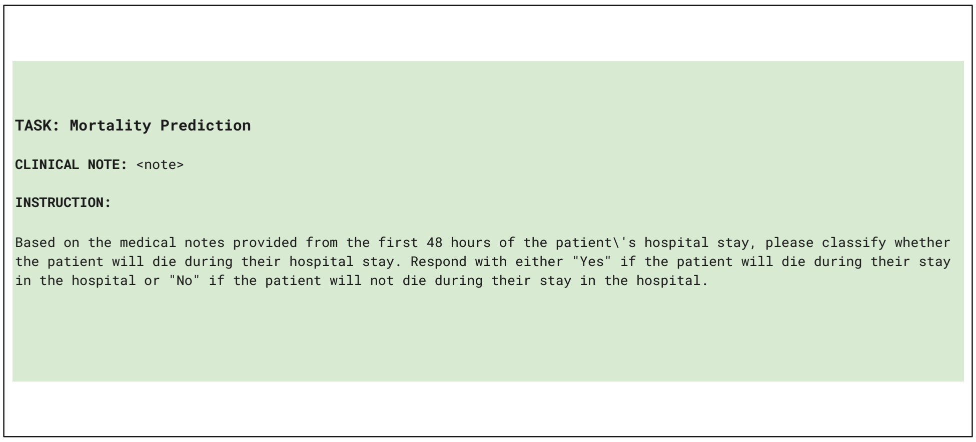
TASK: Cohort Drug Abuse

CLINICAL NOTE: <note>

INSTRUCTION:

Based on the medical records provided, answer the question. Does this patient meet the definition of drug abuse? Respond "Yes" if the patient meets the definition of drug abuse, or respond "No" if the patient does not meet the definition of drug abuse.

Figure 10: Example of instructions for ‘Cohort drug abuse’ classification task.



TASK: Mortality Prediction

CLINICAL NOTE: <note>

INSTRUCTION:

Based on the medical notes provided from the first 48 hours of the patient\'s hospital stay, please classify whether the patient will die during their hospital stay. Respond with either "Yes" if the patient will die during their stay in the hospital or "No" if the patient will not die during their stay in the hospital.

Figure 11: Example of instructions for ‘Mortality Prediction’ classification task.

Model / Dataset	Mistral IT 0.2 (7B)	Llama 2 Chat (13B)	Llama 2 Chat (7B)	Alpaca (7B)	Clinical Camel (13B)	Asclepius (7B)	MedAlpaca (7B)
Obesity Co-Morbidity (Asthma)	0.974 ±(0.014)	0.908 ±(0.111)	0.696 ±(0.145)	0.479 ±(0.017)	0.594 ±(0.059)	0.732 ±(0.086)	0.557 ±(0.078)
Cohort Alcohol Abuse	0.980 ±(0.028)	0.898 ±(0.142)	0.836 ±(0.148)	0.549 ±(0.126)	0.517 ±(0.177)	0.894 ±(0.084)	0.715 ±(0.146)
Obesity Co-Morbidity CAD	0.963 ±(0.017)	0.933 ±(0.067)	0.796 ±(0.096)	0.512 ±(0.033)	0.649 ±(0.107)	0.702 ±(0.154)	0.679 ±(0.071)
Cohort Drug Abuse	0.941 ±(0.039)	0.923 ±(0.04)	0.934 ±(0.048)	0.570 ±(0.132)	0.698 ±(0.138)	0.938 ±(0.042)	0.756 ±(0.119)
Cohort English	0.974 ±(0.055)	0.824 ±(0.123)	0.790 ±(0.165)	0.460 ±(0.071)	0.586 ±(0.076)	0.737 ±(0.078)	0.552 ±(0.058)
Cohort Make Decision	0.709 ±(0.178)	0.623 ±(0.238)	0.710 ±(0.171)	0.644 ±(0.047)	0.597 ±(0.174)	0.817 ±(0.074)	0.513 ±(0.098)
Cohort Abdominal	0.750 ±(0.034)	0.707 ±(0.076)	0.644 ±(0.034)	0.483 ±(0.029)	0.506 ±(0.069)	0.637 ±(0.052)	0.648 ±(0.059)
Obesity Co-Morbidity (Diabetes)	0.987 ±(0.011)	0.958 ±(0.063)	0.775 ±(0.114)	0.560 ±(0.041)	0.637 ±(0.109)	0.762 ±(0.124)	0.686 ±(0.05)
Obesity Classification	0.943 ±(0.05)	0.9 ±(0.087)	0.639 ±(0.113)	0.534 ±(0.03)	0.612 ±(0.074)	0.453 ±(0.177)	0.64 ±(0.084)
Mortality Prediction	0.777 ±(0.034)	0.794 ±(0.036)	0.742 ±(0.083)	0.466 ±(0.051)	0.506 ±(0.052)	0.757 ±(0.037)	0.658 ±(0.08)

Table 3: Mean and Standard Deviation for instructions on classification tasks across all models and all tasks

Model / Dataset	Mistral IT 0.2 (7B)	Llama 2 Chat (13B)	Llama 2 Chat (7B)	Alpaca (7B)	Clinical Camel (13B)	Asclepius (7B)	MedAlpaca (7B)
Medication Extraction	0.351 ±(0.111)	0.559 ±(0.072)	0.608 ±(0.084)	0.231 ±(0.069)	0.509 ±(0.15)	0.562 ±(0.027)	0.529 ±(0.047)
Concept Problem Extraction	0.265 ±(0.051)	0.325 ±(0.035)	0.329 ±(0.027)	0.131 ±(0.029)	0.3 ±(0.035)	0.256 ±(0.019)	0.229 ±(0.021)
Concept Test Extraction	0.154 ±(0.076)	0.197 ±(0.066)	0.236 ±(0.05)	0.097 ±(0.025)	0.117 ±(0.078)	0.194 ±(0.025)	0.109 ±(0.049)
Concept Treatment Extraction	0.165 ±(0.084)	0.244 ±(0.086)	0.367 ±(0.093)	0.086 ±(0.031)	0.198 ±(0.129)	0.308 ±(0.039)	0.193 ±(0.072)
Drug Extraction	0.394 ±(0.101)	0.373 ±(0.047)	0.495 ±(0.072)	0.192 ±(0.074)	0.372 ±(0.128)	0.432 ±(0.042)	0.429 ±(0.086)
Risk Factor CAD Extraction	0.057 ±(0.009)	0.081 ±(0.018)	0.079 ±(0.024)	0.067 ±(0.056)	0.122 ±(0.046)	0.063 ±(0.012)	0.103 ±(0.029)

Table 4: Mean and Standard Deviation for instructions on extraction tasks across all models and all tasks

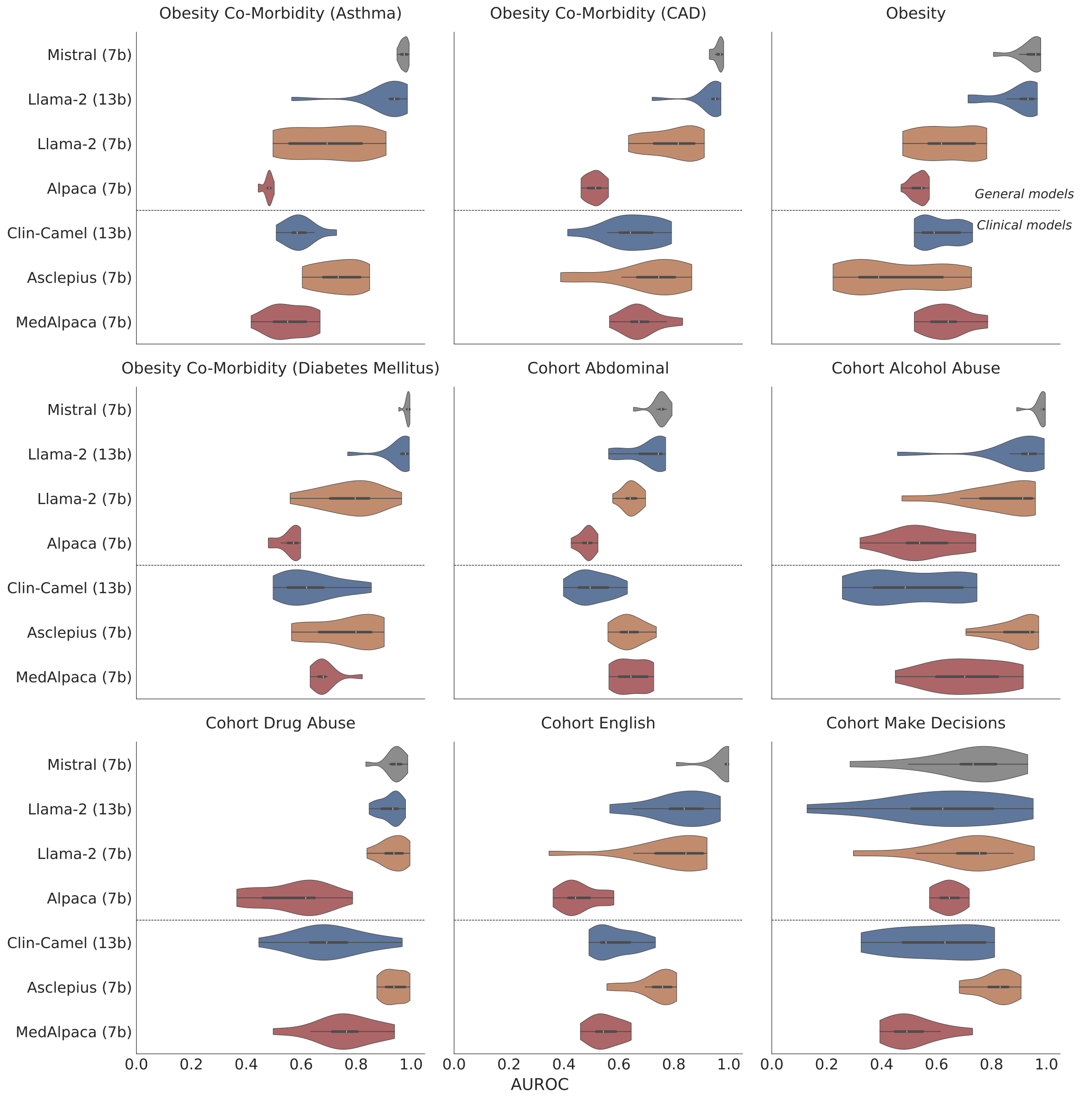


Figure 12: Variability in performance across prompts for binary classification tasks. Again we observe that different (equivalent) instructions yield wide variances in performance, suggesting an undue sensitivity to phrasings.

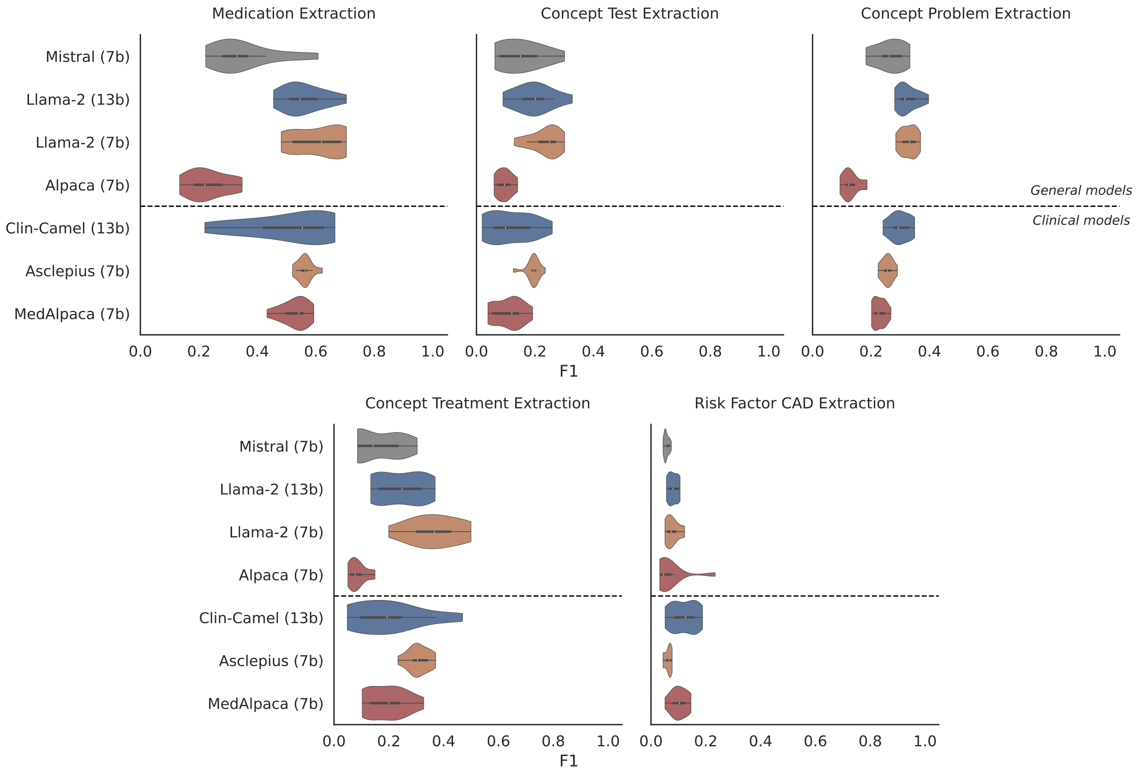


Figure 13: Variability in performance across prompts for the remaining 5 extraction tasks. As mentioned, for most models, different but semantically equivalent prompts yield quite a range of performance.

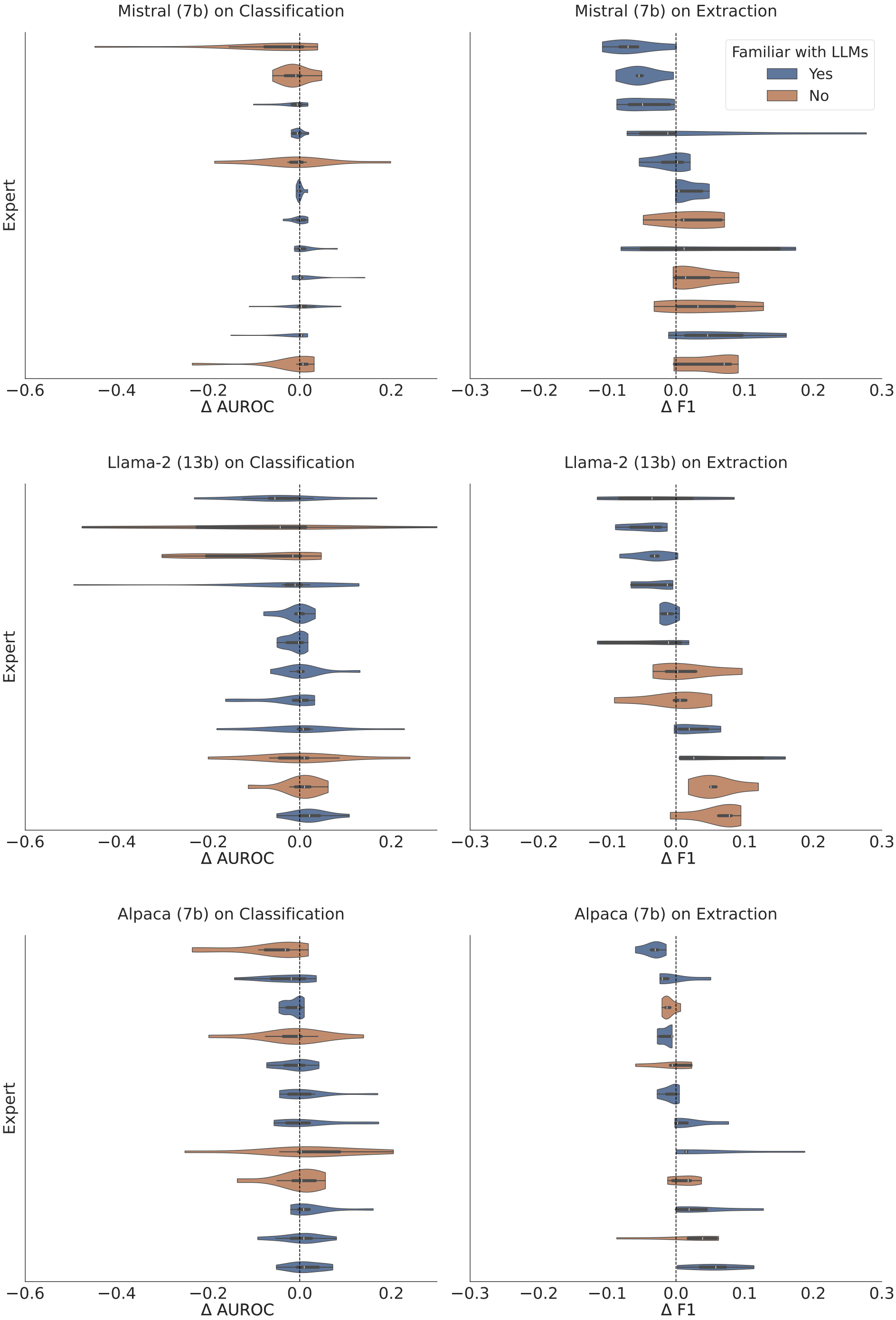


Figure 14: Distribution of performance deltas between each expert's prompt and the median prompt across all tasks for each *general* model. Each violin plot represents an expert color-coded according to their familiarity with LLM.

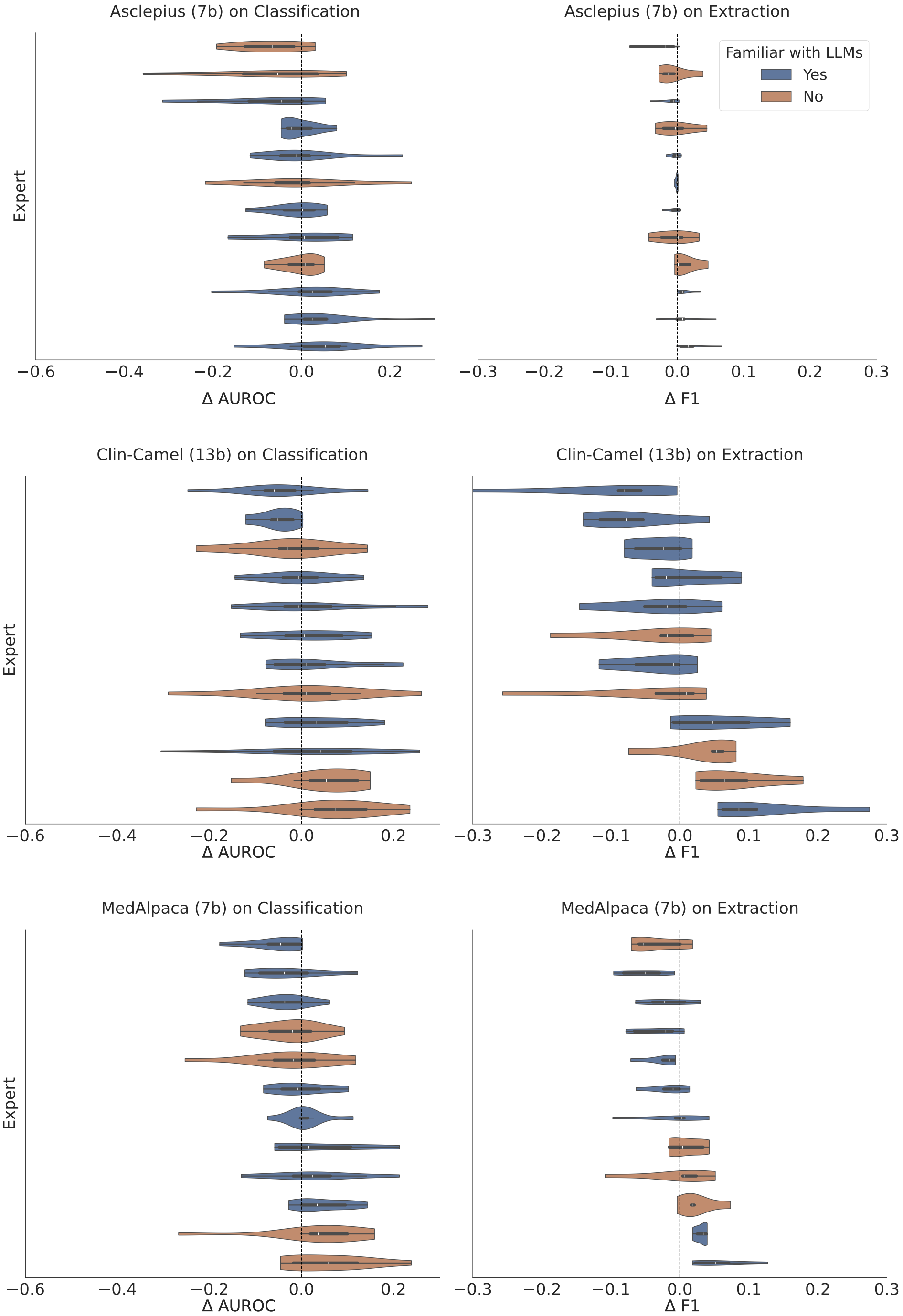


Figure 15: Distribution of performance deltas between each expert's prompt and the median prompt across all tasks for each *clinical* model. Each violin plot represents an expert color-coded according to their familiarity with LLM.

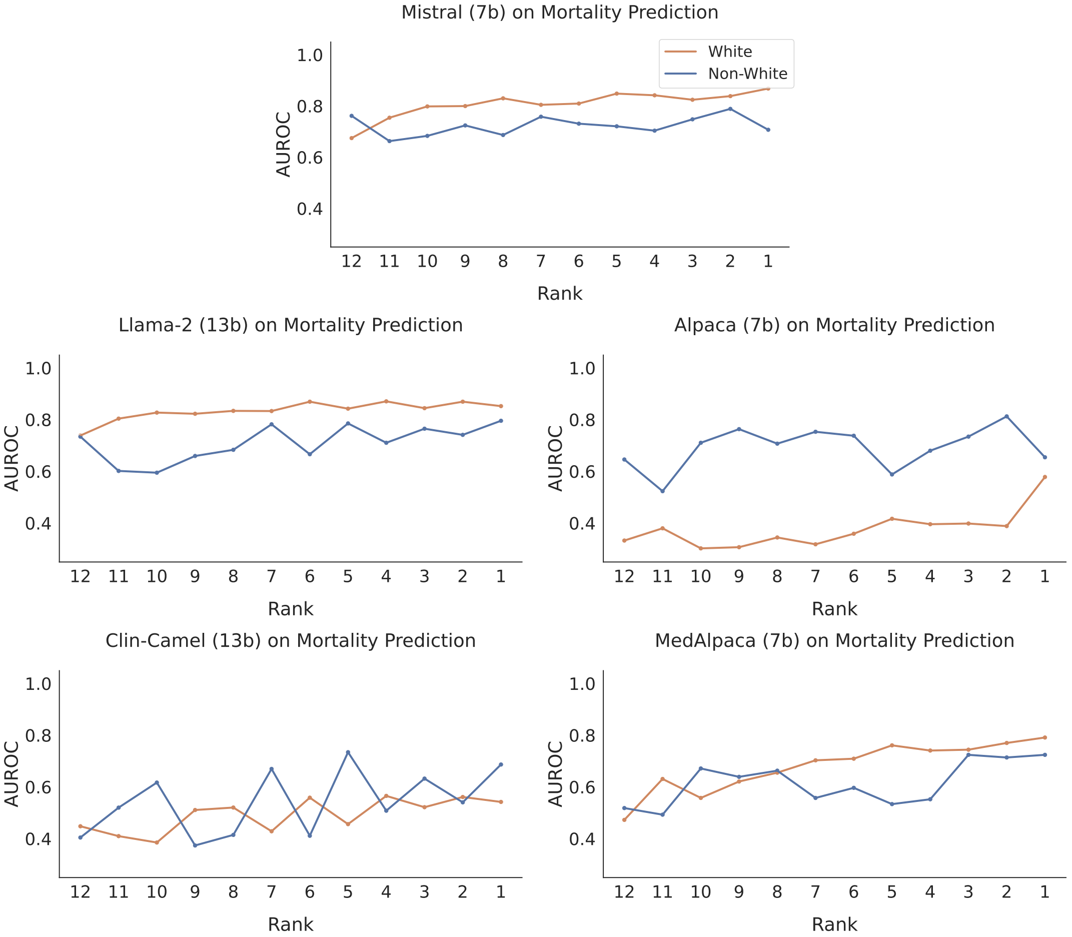


Figure 16: Race subgroup performance on the Mortality Prediction task with a general (left) and clinical model (right). Mistral has no clinical counterpart in our study.

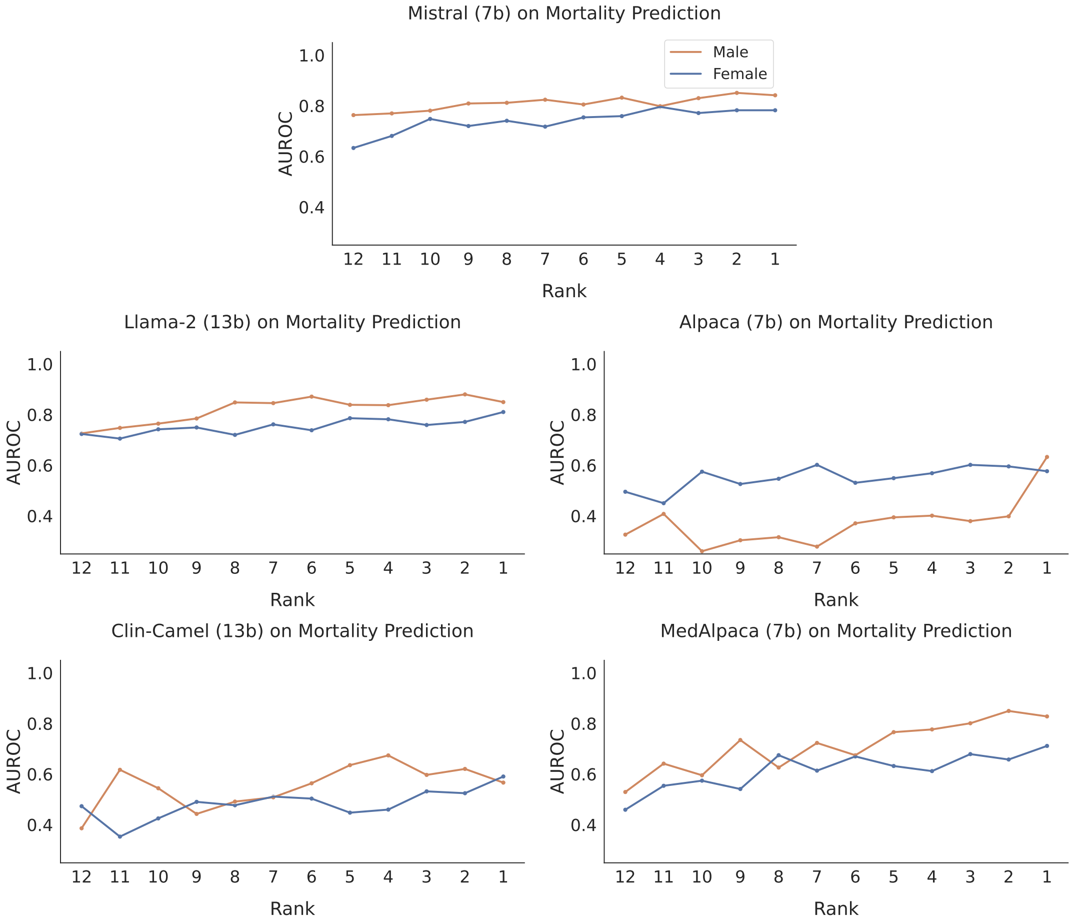


Figure 17: Sex subgroup performance on the Mortality Prediction task with a general (left) and clinical model (right). Mistral has no clinical counterpart in our study.

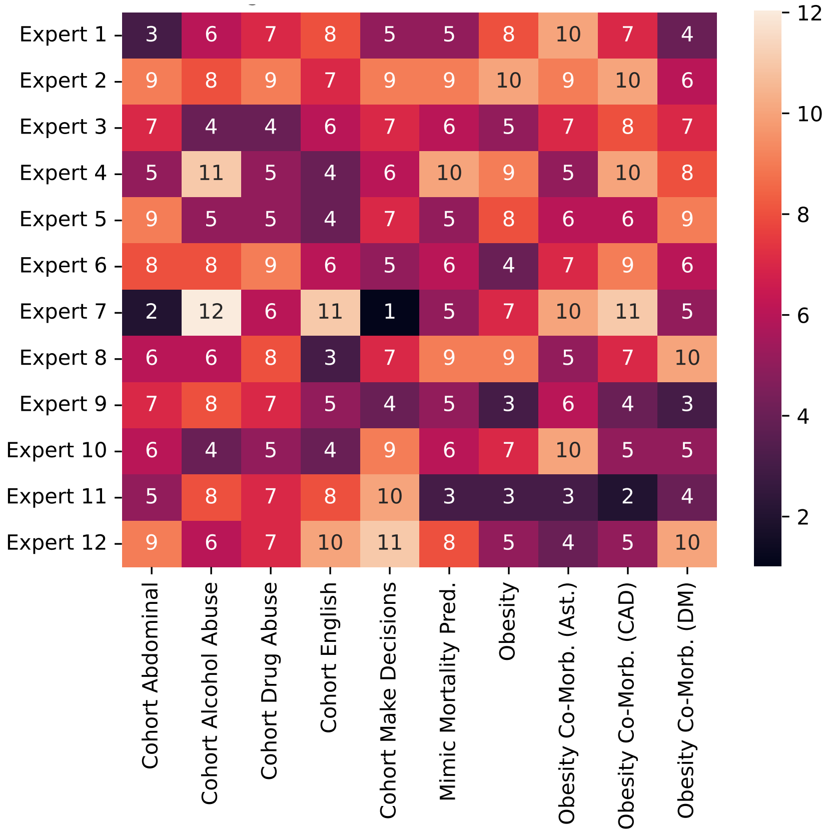


Figure 18: Median ranking of prompts written by experts for classification tasks across models.

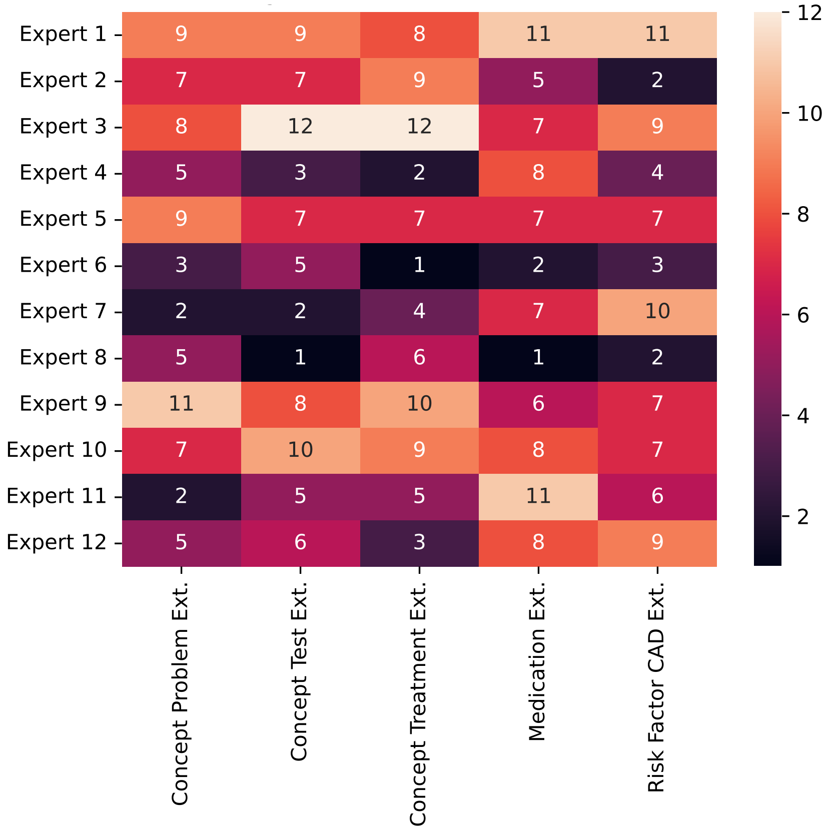


Figure 19: Median ranking of prompts written by experts for extraction tasks across models.

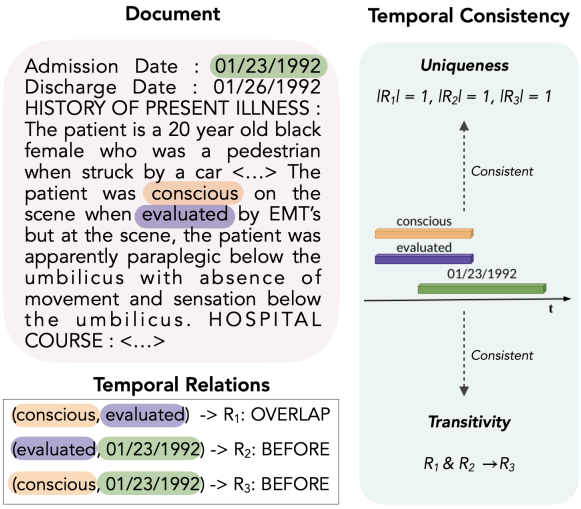
Analysing zero-shot temporal relation extraction on clinical notes using temporal consistency

Vasiliki Kougia[1,2,*], Anastasiia Sedova[1,2], Andreas Stephan[1,2],
Klim Zaporojets[4], Benjamin Roth[1,3]

[1]Faculty of Computer Science, University of Vienna, Vienna, Austria
[2]UniVie Doctoral School Computer Science, Vienna, Austria
[3]Faculty of Philological and Cultural Studies, University of Vienna, Vienna, Austria
[4]Department of Computer Science, Aarhus University, Aarhus, Denmark
[*]vasiliki.kougia@univie.ac.at

Abstract

This paper presents the first study for temporal relation extraction in a zero-shot setting focusing on biomedical text. We employ two types of prompts and five Large Language Models (LLMs; GPT-3.5, Mixtral, Llama 2, Gemma, and PMC-LLaMA) to obtain responses about the temporal relations between two events. Our experiments demonstrate that LLMs struggle in the zero-shot setting, performing worse than fine-tuned specialized models in terms of F1 score. This highlights the challenging nature of this task and underscores the need for further research to enhance the performance of LLMs in this context. We further contribute a novel comprehensive temporal analysis by calculating consistency scores for each LLM. Our findings reveal that LLMs face challenges in providing responses consistent with the temporal properties of uniqueness and transitivity. Moreover, we study the relation between the temporal consistency of an LLM and its accuracy, and whether the latter can be improved by solving temporal inconsistencies. Our analysis shows that even when temporal consistency is achieved, the predictions can remain inaccurate.



Admission Date : 01/23/1992
Discharge Date : 01/26/1992
HISTORY OF PRESENT ILLNESS : The patient is a 20 year old black female who was a pedestrian when struck by a car <...> The patient was conscious on the scene when evaluated by EMT's but at the scene, the patient was apparently paraplegic below the umbilicus with absence of movement and sensation below the umbilicus. HOSPITAL COURSE : <...>

Temporal Relations

(conscious, evaluated) -> R_1: OVERLAP
(evaluated, 01/23/1992) -> R_2: BEFORE
(conscious, 01/23/1992) -> R_3: BEFORE

Figure 1: An example of three event pairs annotated with temporal relations. In the right part, the order of the events with respect to time (t) is shown and the consistency of uniqueness and transitivity.

1 Introduction

Reasoning regarding the temporality of events detected in a text (e.g., understanding their duration, frequency, and order) is an essential part of natural language understanding (Allen, 1983; Wenzel and Jatowt, 2023). Event ordering can be approached as identifying temporal relations between two events, a task often referred to as *temporal relation extraction* (TempRE). This task can also be applied to medical text (BioTempRE), e.g., clinical notes written by clinicians regarding a patient's visit, and various medical events such as symptoms, treatments, tests, and other medical terms (see Figure 1). BioTempRE has numerous useful applications in healthcare and can assist in medical diagnosis, including adverse drug event detection and medical history construction (Sun et al., 2013; Gumiel et al., 2021; Haq et al., 2021; Tu et al., 2023). Current state-of-the-art methods perform supervised learning, which requires annotated datasets (Wang et al., 2022; Yao et al., 2022; Knez and Žitnik, 2024). However, acquiring high-quality annotated data for TempRE poses significant challenges causing problems to existing datasets like missing relations and low inter-annotator agreement (Ning et al., 2017). In the biomedical domain, this challenge is aggravated by the need for expert knowledge and the sensitive nature of medical data.

In TempRE, there are important properties that emerge from the temporal nature of this task and determine the relations between events (see Figure 1). Such properties are *symmetry* (e.g., A *BEFORE* B ⇒ B *AFTER* A) and *transitivity* (e.g., A *BEFORE* B and B *BEFORE* C ⇒ A *BEFORE* C). Additionally, we identify the property of *uniqueness*: each

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 72–84
August 16, 2024. ©2024 Association for Computational Linguistics

pair of events can have only one temporal relation since they are mutually exclusive. These properties can be utilized to enforce global *temporal consistency* on a model's predictions: for example, on a unified output of different classifiers (Chambers et al., 2014; Tang et al., 2013), on a model that operates locally (i.e., with one pair of events as input, Ning et al. (2017)), or on predicted relations between different types of events (Wang et al., 2022).

Recently, Large Language Models (LLMs) have shown remarkable performance in several tasks even in a zero-shot setting, which helps to tackle the need for training data (Bubeck et al., 2023; Wei et al., 2022a). Numerous works experiment with predictions of LLMs and study their reasoning abilities and the impact of various prompts in different tasks (Wu et al., 2023b; Jain et al., 2023; Tan et al., 2023). Despite the success of LLMs, studies show that these models continue to face challenges in temporal reasoning, especially in TempRE (Tan et al., 2023; Jain et al., 2023; Yuan et al., 2023), as well as in biomedical tasks (Wu et al., 2023b). In zero-shot TempRE, Yuan et al. (2023) employed different prompts for ChatGPT and found that it has a considerably lower performance compared to standard supervised methods. They also report ChatGPT's tendency to provide temporally inconsistent responses, but only in terms of symmetry and did not perform an evaluation of temporal consistency specifically. Furthermore, to the best of our knowledge, we are the first ones to investigate the temporal reasoning capabilities of LLMs on medical data.

In this paper, we perform zero-shot BioTempRE on clinical notes (i.e., medical texts documenting patients visits) by using prompts consisting of a clinical note and questions regarding which temporal relation exists between a pair of events.[1] We experiment with two different prompting strategies (BatchQA and CoT) and five widely-used LLMs (GPT-3.5, Mixtral 8x7B, Llama2 70B, Gemma 7B, and PMC-LLaMA 13B). Our findings reveal that LLMs perform poorly in this task, with a difference of approximately 0.2 in F1 score compared to supervised models. Furthermore, we calculate consistency scores for uniqueness and transitivity for each LLM in order to assess their temporal consistency and its impact on accuracy. Consistency is later enforced on the predictions with an Integer Linear Programming (ILP) method, revealing that solving the inconsistencies does not improve the F1 score.

Overall, our contributions are:

- To the best of our knowledge, this is the first study of zero-shot BioTempRE.
- We provide extensive quantitative results of two types of prompts and five different LLMs.
- We perform a novel temporal consistency analysis by calculating consistency scores for temporal properties.
- We study how temporal consistency relates to accuracy and enforce it using an ILP method.
- The code and data containing the prompts, the raw and the processed responses by the LLMs for around 600,000 pair instances, will be publicly shared for further analysis.[2]

2 Related Work

2.1 Temporal Relation Extraction

Multiple studies on addressing TempRE have applied temporal properties to classifiers' predictions, either during training or at inference time, aiming to improve their performance (Tang et al., 2013; Chambers et al., 2014; Ning et al., 2017, 2018; Wang et al., 2022). Other works have also employed linguistic properties or properties based on causality (Chambers et al., 2014; Ning et al., 2018). Ning et al. (2018) formulated temporal, causal, and linguistic properties as constraints for an ILP method. Later, Liu et al. (2021) showed that ILP constraints can improve temporal consistency, although in certain cases, the F1 score may decrease.

Temporal Relation Extraction in the Medical Domain. The 2012 Informatics for Integrating Biology and the Bedside (i2b2) challenge was the first to address the BioTempRE task (Sun et al., 2013). The best-performing method involved merging predictions from different SVM and CRF classifiers with regard to temporal consistency (Tang et al., 2013). Following challenges at SemEval, called Clinical TempEval, were organized from 2015 to 2017 (Bethard et al., 2015, 2016, 2017) and utilized the THYME corpus (Styler IV et al.,

[1] We do not perform event detection and instead consider the events in each text already known.

[2] https://github.com/vasilikikou/consistent_bioTempRE

2014).[3] In 2015 and 2016, the best-performing methods were CRF- and SVM- based (Velupillai et al., 2015; Lee et al., 2016; Khalifa et al., 2016), while in 2017 the winning approach employed an LSTM (Tourille et al., 2017). Following approaches have utilized BERT (Lin et al., 2019; Haq et al., 2021; Tu et al., 2023) for relation classification given a text and an event pair. Recently, Knez and Žitnik (2024) introduced a multimodal method in which, they constructed a graph with medical information and then, they combined textual representations (extracted by BERT) and graph representations (extracted by a GNN). However, even though temporal consistency has been used in existing TempRE works, it has not been utilized for analyzing the performance of a model by calculating consistency scores.

2.2 Zero-Shot Temporal Relation Extraction

Zero-shot learning (Xian et al., 2019) enables models to execute tasks without explicit training, a capability demonstrated by scaling models since GPT-3 (Brown et al., 2020; Wei et al., 2022a). Instruction tuning techniques (Wei et al., 2022a) further enhance zero-shot learning in LLMs. Recent openly available LLMs like LLama (Touvron et al., 2023) and Mixtral (Jiang et al., 2024) narrow the gap with closed-source models, while chain-of-thought (CoT) prompting (Wei et al., 2022b) has enhanced their ability to handle complex tasks. Research studies have shown that the temporal reasoning tasks remain challenging for LLMs (Jain et al., 2023), and specifically for TempRE, where Yuan et al. (2023) explored zero-shot TempRE with ChatGPT and found that it yields a large performance gap compared to supervised methods. However, previous research has not analyzed zero-shot TempRE in the medical domain or the temporal consistency and its impact on the performance of zero-shot TempRE - both gaps we aim to fill in our work. In this paper, we calculate consistency scores and study their connection to the F1 scores.

3 Methodology

3.1 Problem Formulation

Given a text document D and a set of events $E = \{e_1, .., e_{|E|}\}$ mentioned in the text, we create pairs of events, which are represented by the set $P = \{p_1, .., p_i, .., p_{|P|}\}$, where p_i indicates the i^{th} pair, $1 \leq i \leq |P|$. BioTempRE aims at assigning the appropriate temporal relation r to the corresponding pair of events. Each $p_i \in P$ is described by two distinct events $\{e_j, e_k\}$, where $1 \leq j, k \leq |E|$. Furthermore, each event $e \in E$ is characterized by the points in time at which it began and finished. These temporal points are denoted as b and f, respectively.

Following the work of Ning et al. (2018), we employ the same relation scheme, which consists of 5 different types of temporal relations r: *before*, *after*, *includes*, *is included*, and *simultaneously*, represented by the label set $R_T = \{r_B, r_A, r_I, r_{II}, r_S\}$. We choose this set of relations based on the fact that they are fine-grained and well-defined, and hence, suitable for creating temporal rules for our analysis. An r_B temporal relation indicates that $b(e_j) < b(e_k)$ and $f(e_j) < f(e_k)$, while an r_A temporal relation signifies that $b(e_j) > b(e_k)$ and $f(e_j) > f(e_k)$. Furthermore, r_I indicates that $b(e_j) \leq b(e_k)$ and that $f(e_j) \geq f(e_k)$, and r_{II} signifies that $b(e_j) \geq b(e_k)$ and that $f(e_j) \leq f(e_k)$. Finally, r_S signifies that $b(e_j) = b(e_k)$ and $f(e_j) = f(e_k)$.

3.2 Zero-shot BioTempRE

We experiment with two different types of prompting: *Batch-of-Questions* (BatchQA) and *Chain-of-Thought* (CoT) (Wei et al., 2022b; Yuan et al., 2023) (see Figure 4 in Appendix A). In both, we start with a preamble consisting of the document text (D) and an instruction. Then, we introduce questions regarding the temporal relations for a pair of events p_i consisting of events e_j and e_k.[4] We formulate the question for each relation based on its temporal definition, as follows:

- *BEFORE*: Did e_j start before e_k started and end before e_k ended?
- *AFTER*: Did e_j start after e_k started and end after e_k ended?
- *INCLUDES*: Did e_k start and end while e_j was happening?
- *IS INCLUDED*: Did e_j start and end while e_k was happening?
- *SIMULTANEOUS*: Did e_j and e_k start and end at the same time?

[3] The i2b2 dataset is publicly available. The THYME corpus is provided upon request, however our requests were not answered.

[4] The questions were ordered randomly.

We also specify the desired output format by adding *"Answer with Yes or No."* to the end of each question. For each event pair there is an independent interaction with the LLM, and depending on the type of prompt the questions mentioned above are sent to the LLM in one or multiple prompts.

Batch-of-Questions (BatchQA). In BatchQA, a single prompt is sent to the LLM. In the preamble, after the document D, this instruction follows: *"Given document D, answer the following questions ONLY with Yes or No."*. Next, all the questions regarding the temporal relations are added in the same prompt. The expected model response includes five *Yes/No* answers for each of the questions.

Chain-of-Thought (CoT). We use the same format of temporal prompts as in Yuan et al. (2023) (based on their examples in the paper), and we formulate the questions for the set of the 5 temporal relations used in Ning et al. (2018). The first prompt is the preamble composed of the document D and the question *"Given the document D, are e_j and e_k referring to the same event? Answer ONLY with Yes or No."*. If the response is *No*, then the questions are sent, each one after another, as they are defined above. If the response is *Yes*, the phrase *"In that event,"* is appended at the beginning of each question.

4 Experimental Setup

4.1 Data

In our experiments, we use the dataset created for the 2012 i2b2 challenge, which consists of 310 discharge summaries, 190 for training and 120 for testing. The texts were initially annotated with 8 fine-grained relations but due to low inter-annotator agreement these relations were merged to the following three: *before, after and overlap*. Each discharge text contains 30.8 sentences on average, with each sentence having an average number of 17.7 tokens. The average number of tokens per discharge text is 514.

The i2b2 dataset contains three types of events: 1) medical events, 2) time expressions, and 3) the dates of admission and discharge. The average number of medical events per discharge summary is 86.7, while the average number of time expressions is 10.5. The admission and discharge dates are included in each text; however, in a few cases, one of them might be missing. The annotators of i2b2 have assigned temporal relations to 27,540 pairs of events (gold pairs).

An important step in TempRE is to identify the pairs of events for which the models will decide if there is a relation expressed or not since it would not be feasible to check for every pair of events mentioned in a document. In order to generate candidate event pairs, we follow the approach of the best-performing method in the i2b2 challenge (Tang et al., 2013). This is a rule-based approach, which creates pairs consisting of every event and the admission and discharge dates, every two consecutive events within the same sentence, and events in the same as well as in different sentences based on linguistic rules. The generated candidate pairs are 60,840 in total, from which 28.16% appears also in the gold pairs.

The five relations we use in our experiments (see Section 3) are different from the gold ones existing in the dataset. In order to evaluate the prediction of our methods, we map the five relations to the three gold ones as follows: before → before, after → after, includes → overlap, is included → overlap and simultaneously → overlap.

4.2 Methods

LLMs We employ the following five (one closed-source and four open-weight) models of various sizes: GPT-3.5 ("ChatGPT"),[5] Gemma 7B (Team et al., 2024),[6] Mixtral 8x7B (Jiang et al., 2024),[7] Llama2 70B (Touvron et al., 2023),[8] and PMC-LLaMA 13B, which is pre-trained on medical text (Wu et al., 2023a).[9] PMC-LLaMA is only instruction-tuned on QA data (respond to one question at a time) and thus does not follow the format of BatchQA prompts, which expect multiple outputs. Therefore, we use it only for CoT. The experiments were costly in terms of time (and money for GPT-3.5), especially for CoT, where each question is sent separately. The running times ranged from three hours (Gemma BatchQA) to 7 days (Llama CoT) (see more details in Appendix A).

Baselines We implement a rule-based baseline, called W-order, where only the *before* and *after*

[5] https://openai.com/index/gpt-3-5-turbo-fine-tuning-and-api-updates/

[6] https://huggingface.co/google/gemma-1.1-7b-it

[7] https://huggingface.co/mistralai/Mixtral-8x7B-Instruct-v0.1

[8] https://huggingface.co/meta-llama/Llama-2-70b-chat-hf

[9] https://huggingface.co/axiong/PMC_LLaMA_13B

relations are predicted for each event pair based on the order in which the events are mentioned in the text. A combination of the predictions of each LLM with the W-order predictions is also implemented. In cases where the LLM gives a negative or uncertain prediction for all the relations, the prediction of W-order is used instead.

5 Zero-shot TempRE results

To evaluate the correctness of the predicted relations, we calculate the precision, recall and F1 scores. For each pair of events $p_i = (e_j, e_k)$, we check if the predicted relation r_i matches the gold relation. Hence, we calculate the triple (e_j, e_k, r_i) match between the predictions and the ground truth. In Table 1, the results for the gold and candidate pairs are presented. In order to perform a fair comparison, considering that not every candidate pair of events has a gold annotation (and therefore it is unknown whether a prediction is correct or wrong), we only evaluate those generated candidate pairs that are also contained in the gold pairs. If a gold pair does not exist in the generated candidate pairs, there is no prediction for it, and that would affect the recall score negatively. In the Supervised setting, we show scores reported by the corresponding papers. Knez and Žitnik (2024) do not mention event detection or candidate pair generation, hence we assume they used the gold pairs. On the other hand, we show the results from Haq et al. (2021) and Tu et al. (2023) in the Candidate column since they operate on events they have detected in the text.

Our experiments demonstrate that the best performing methods are the same for the gold and the candidate pairs. As expected, the F1 score of the methods when the candidate pairs are used is lower, mostly due to the decrease in recall. The best performing method is *Llama CoT + W-order* in terms of F1 score. On the other hand, *Mixtral CoT* achieves the best precision score and *Gemma BatchQA + W-order* the best recall. Overall, the supervised methods consistently outperform the methods in the zero-shot setting, with an average difference of approximately 20% F1 score. In general, most LLMs (except for *Gemma*) exhibit improved performance when the CoT prompting approach is used. However, in an LLM-based comparison, we observe that the performance varies depending on the type of prompt used. For example, Llama with CoT has the highest F1 score, but when BatchQA is used, the score drops almost in half. Moreover, the combination of *W-order* predictions with the zero-shot methods yields improvements in recall and F1 score, but in most cases, it harms precision. Notably, *PMC-LLaMA*, the medical LLM we employed, has low results and is often outperformed by the general domain LLMs, showing no advantage from pre-training on biomedical text.

6 Temporal consistency analysis

Considering the temporal nature of the TempRE task, we investigate the impact of incorporating the following two temporal properties in the zero-shot setting: 1. *uniqueness*, requiring that each event pair has exactly one relation, and 2. *transitivity* (see transitivity rules in Table 4 in the Appendix). First, we evaluate the zero-shot methods based on their consistency, i.e., if their predictions follow the temporal properties or not. Then, we use ILP to enforce temporal consistency on the predictions. Specifically, we examine the following three questions:

- How consistent are different zero-shot methods?
- How is the temporal consistency of the predictions connected to their correctness?
- Can the predictions be improved by a temporal constraint-based ILP method?

How consistent are different zero-shot methods? We calculate two consistency scores: one for uniqueness c_U and one for transitivity c_T, which show the percentage of cases where the corresponding temporal property was not violated. The consistency score for uniqueness is calculated based on the number of pairs as follows:

$$c_U = \frac{\sum_{i=1}^{P} p_{i,|r_i|=1}}{|P|}, \tag{1}$$

where only the pairs p_i with a singular predicted relation r_i are considered. In Table 2, the consistency scores for uniqueness are reported. Furthermore, we present the number of pairs for which no relation was predicted (# 0) and the number of pairs with more than one predicted relation (# >1). We observe that all the models struggle to keep consistency, especially because of assigning more than one relation to a pair. For the majority of the evaluated LLMs, this occurs for at least 50% of the pairs

Setting	Method	Gold			Candidate		
		P	R	F1	P	R	F1
Rule-based	W-order	0.348	0.348	0.348	0.382	0.305	0.339
Supervised	Haq et al. (2021)†	-	-	-	-	-	0.736
	Tu et al. (2023)†	-	-	-	0.645	0.672	0.650
	Knez and Žitnik (2024)	-	-	**0.820**	-	-	-
	Tang et al. (2013)	-	-	-	0.714	0.673	**0.693**
Zero-shot	GPT-3.5 BatchQA	0.588	0.083	0.132	0.607	0.060	0.101
	GPT-3.5 BatchQA + W-order	0.395	0.397	0.396	0.424	0.340	0.377
	GPT-3.5 CoT	0.400	0.641	0.491	0.387	0.494	0.432
	GPT-3.5 CoT + W-order	0.400	0.677	0.502	0.390	0.528	0.447
	Mixtral BatchQA	0.458	0.534	0.491	0.420	0.392	0.404
	Mixtral BatchQA + W-order	0.452	0.572	0.504	0.422	0.428	0.424
	Mixtral CoT	**0.681**	0.504	0.576	**0.694**	0.422	0.520
	Mixtral CoT + W-order	0.545	0.596	0.569	0.561	0.494	0.524
	Llama BatchQA	0.366	0.371	0.367	0.316	0.254	0.281
	Llama BatchQA + W-order	0.367	0.411	0.387	0.327	0.292	0.308
	Llama CoT	0.549	0.710	0.615	0.551	0.567	0.555
	Llama CoT + W-order	0.534	0.742	**0.620**	0.538	0.595	**0.564**
	Gemma BatchQA	0.426	0.837	0.564	0.425	0.667	0.519
	Gemma BatchQA + W-order	0.426	**0.838**	0.565	0.425	**0.668**	0.519
	Gemma CoT	0.429	0.398	0.401	0.449	0.318	0.358
	Gemma CoT + W-order	0.385	0.552	0.452	0.407	0.458	0.429
	PMC-LLaMA CoT	0.395	0.516	0.439	0.406	0.425	0.408
	PMC-LLaMA CoT + W-order	0.390	0.574	0.463	0.403	0.476	0.435

Table 1: Precision (P), recall (R) and F1 scores of TempRE methods on the gold and candidate pairs. Methods with † use a different candidate pair generation than ours, so their results are not directly comparable to ours.

Method	Gold				Candidate			
	c_U (%)	# 0	# >1	c_T (%)	c_U (%)	# 0	# >1	c_T (%)
GPT-3.5 BatchQA	8.03	24,476	860	70.34	5.06	56,457	1,306	68.78
GPT-3.5 CoT	13.07	2,657	21,284	46.58	14.91	6,194	45,573	47.29
Mixtral BatchQA	59.94	3,102	7,931	71.20	60.13	7,192	17,063	71.87
Mixtral CoT	37.60	10,315	6,868	68.99	35.22	27,434	11,980	67.44
Llama BatchQA	**71.67**	2,858	4,945	**82.35**	**70.58**	6,451	11,446	**80.05**
Llama CoT	30.55	2,916	16,211	60.39	33.64	6,864	33,507	59.45
Gemma BatchQA	2.67	57	26,747	63.04	2.26	115	59,347	62.59
Gemma CoT	3.82	14,159	12,335	60.00	3.00	32,605	26,411	60.56
PMC-LLaMA CoT	33.18	7,469	10,933	60.45	31.88	17,977	23,465	59.85
W/ ILP reasoning	100	0	0	100	100	0	0	100

Table 2: Temporal consistency scores for uniqueness (c_U) and transitivity (c_T) for each model. The consistency scores show the percentage of pairs which are consistent for the corresponding property. # 0 and # >1 shows the number of pairs with none and more than one predictions respectively.

and can go up to 97% (Gemma BatchQA). In this evaluation, we also find that there is no clear winner among the LLMs or the prompt types, since the same LLM shows different levels of consistency with different prompt types. The combination with the highest consistency for uniqueness is Llama with BatchQA.

The consistency score for transitivity is calculated based on triples of event pairs in the following form: $((e_i, e_j), (e_j, e_k), (e_i, e_k))$. We first find these triples in the dataset and then obtain the relations predicted for them. If r_1, r_2 and r_3 are the predictions for each respective pair in the triple, then for r_3, it should hold that $r_3 \in trans(r_1, r_2)$.[10] If it does not hold, then we have a transitivity violation. Therefore c_T is calculated as:

$$c_T = \frac{\sum_{i=1}^{|Tr|} t_{i, r_3 \in trans(r_1, r_2)}}{|Tr|}, \tag{2}$$

where Tr is the set of transitivity triples and, for each triple t_i, the transitivity for the predicted relations holds.[11] Table 2 showcases the c_T scores for each of the evaluated methods. Similar to *uniqueness*, Llama BatchQA demonstrates the highest consistency for *transitivity*. In general, for all LLMs, we observe that the BatchQA approach yields higher transitivity consistency scores than CoT.

How is the temporal consistency of the predictions connected to their correctness? When comparing the consistency scores with F1, we observe a contradiction. Models that have high consistency have a lower F1 score. In particular, Llama BatchQA is the most consistent in terms of uniqueness and transitivity, but has one of the lowest F1 scores. Especially for the candidate pairs, the F1 score is even lower than the rule-based baseline, yet c_U is 70.58% and c_T is 80.05%. Moreover, Llama CoT, which is the best in terms of F1 score, has low consistency with around only 30% of predictions being unique and 60% correct transitivity triples. These insights suggest that temporal consistency does not always mean correctness.

Can the predictions be improved by a temporal constraint-based ILP approach? Following the approach proposed by Ning et al. (2017, 2018), we implemented an ILP step that uses the temporal properties as constraints and changes inconsistent predictions so that the constraints are not violated.[12] This study aims to investigate whether enforcing consistency can improve the accuracy of the predictions. First, we assign a confidence score sc to each triple $(e_i, e_j, r_k), \forall r_k \in R_T$. The score sc for a pair of events $p = (e_i, e_j)$ equals 1, if the relation was predicted from the model, and 0.2 otherwise. Next, we create a binary vector, which is optimized with ILP. We refer to it as the indicator $I(p_i, r_i) \in [0, 1], \forall p \in P, r \in R_T$. We formulate the constraints as follows:

- Uniqueness:

$$\sum_{p \in P, r \in R_T} I(p, r) = 1 \tag{3}$$

- Symmetry:

$$I(p_i, r_i) = I(p_i^s, \bar{r}_i) \tag{4}$$

  where $p_i = (e_i, e_j)$ and $p_i^s = (e_j, e_i)$, and $\bar{r}_i$ is the reverse relation of r_i.[13]

- Transitivity:

$$I((e_i, e_j), r_1) + I((e_j, e_k), r_2) - \sum_{r_3 \in tr(r_1, r_2)} \leq 1 \tag{5}$$

  where $r_1, r_2, r_3 \in R_T$ and $trans(r_1, r_2)$ is the set of relations that are the transitive of relations r_1 and r_2.

The objective of the ILP method is to maximize the confidence score sc based on the indicator I:

$$\hat{I} = \arg\max \sum_{p \in P} \sum_{r \in R_T} sc(p, r) I(p, r) \tag{6}$$

As shown in Table 2, when the ILP reasoning step is applied, the consistency scores for both uniqueness and transitivity reach 100%. We applied this step to the predictions of Llama BatchQA and Llama CoT, which are the models with the highest contradiction between F1 and consistency. In Table 3, we show the results before and after applying the temporal constraints. Even though the consistency of the predictions after reasoning is 100%, the F1 score decreases slightly for BatchQA and by 0.066 for CoT. This means that the predictions are temporally consistent, but they are not accurate. To get a better understanding of this issue, Figure 2 demonstrates two examples of transitivity triples for which the predictions violate the transitivity constraint. This indicates that at least one of the three predictions is incorrect and needs to change. In the top example, the first two relations were correct, but these relations were changed by the ILP step, resulting in only one relation being

[10] The transitive relations for the relation set we used can be found in Table 4 in Appendix A.

[11] Triples where at least one pair was not assigned a relation are excluded from this calculation.

[12] For the ILP implementation we used the Gurobi optimizer (https://www.gurobi.com/solutions/gurobi-optimizer/).

[13] The reverse of each relation can be found in Table 5 in Appendix A.

Method	W/o ILP reasoning			W/ ILP reasoning		
	P	R	F1	P	R	F1
Llama BatchQA	0.366	0.371	0.367	0.366	0.366	0.366
Llama CoT	0.549	0.710	0.615	0.549	0.549	0.549

Table 3: Precision (P), recall (R) and F1 scores before and after the ILP temporal reasoning step on the gold pairs.

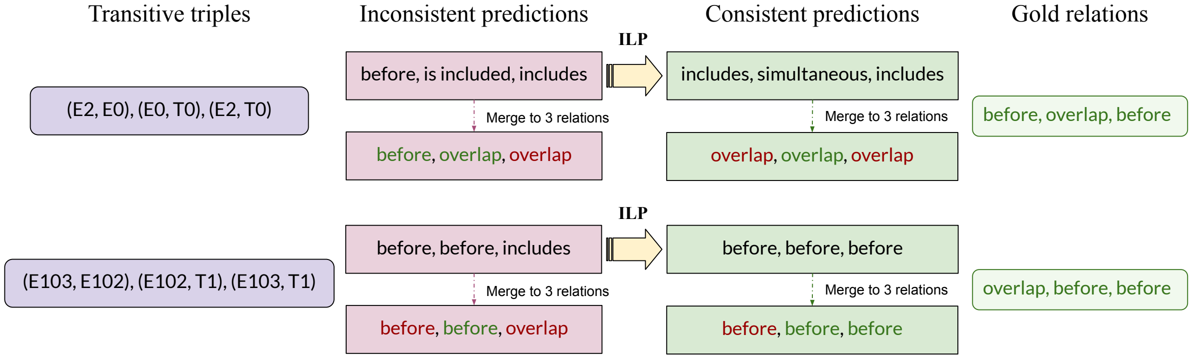


Figure 2: Examples of two transitive triples with inconsistent predictions. After the ILP the predictions are consistent but still different from the gold relations.

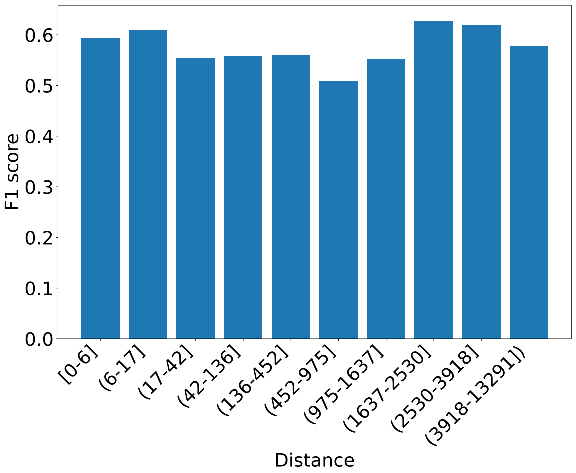


Figure 3: Barplot where each bar represents a range of distances between events in the gold pairs. The y axis shows the F1 score of the predictions for the pairs in each bar.

correct in the consistent predictions. In the bottom example, only one relation was changed to enforce consistency. This resulted in two correctly predicted relations after the ILP, but still the first relation remained incorrect. This analysis highlights our previous observation regarding the relation between consistency and accuracy, and points out to the need of aligning these two aspects more effectively in order for models to achieve an improved performance in temporal reasoning.

7 Pairs distance-based analysis

Since clinical notes contain long texts (see Section 4.1), we perform an analysis based on the distance of event pairs for the best-performing LLM (Llama CoT). First, we calculate the distance in terms of characters between the events for all the gold pairs. Then, we sort the pairs by their distances and split them to 10 bins, so that each bin contains roughly the same amount of pairs. Finally, the F1 score is calculated for the prediction of the pairs contained in each bin. Figure 3 depicts the barplot with the bars representing the pairs in the specific distance range and the corresponding F1 scores. We observe that 37.5% of the pairs have a distance of 0 to 100 characters. Larger distances appear less frequently and hence the range of distance is smaller in the first bars, while the last bars have larger ranges. There is no consistent decrease in F1 score as the distance increases, meaning that the model is not affected by the distance of events in the text.

8 Conclusion

In this paper, we performed BioTempRE on clinical notes in a zero-shot setting employing five different LLMs. Two types of prompts were used, namely BatchQA and CoT, in order to obtain LLMs' responses. The zero-shot performance of all LLMs is lower compared to supervised learning methods. Moreover, we perform a temporal evaluation by calculating the consistency score of each LLM for the temporal properties of uniqueness and transivity. We find that, in general, LLMs' predictions are temporally inconsistent and, interestingly, the model with the higher consistency scores (Llama

BatchQA) has one of the lowest F1 scores. An ILP method utilized to enforce consistency on the models' predictions fails to improve their accuracy. These findings indicate the importance of the relation between temporal consistency and correctness, emphasizing the need for further study in order to assist temporal reasoning.

Acknowledgments

This research has been funded by the Vienna Science and Technology Fund (WWTF)[10.47379/VRG19008] "Knowledge-infused Deep Learning for Natural Language Processing".

Limitations

The gold relations annotated in the dataset are only three, coarse-grained and not well-defined with respect to when they start and end. The consistency analysis we performed is based on rules, which are connected to the definition of relations and their starting and end points. So in order to make sure that the consistency is calculated accurately, we used a set of 5 well-defined fine-grained relations. However, for evaluating the results we need to map the 5 relations to the original set of 3. This, in some cases, could lead to an inaccurate comparison between the gold and the predicted relations. Also, for the prompts, we used only the set of questions mentioned in Section 3.2 and did not perform any prompt tuning. Experimenting with different ways of formulating the questions could help in finding prompts that yield better results. Another research direction could be to add instructions to the prompts for uniqueness and transitivity towards obtaining consistent predictions.

References

James F Allen. 1983. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843.

Steven Bethard, Leon Derczynski, Guergana Savova, James Pustejovsky, and Marc Verhagen. 2015. SemEval-2015 task 6: Clinical TempEval. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 806–814, Denver, CO, USA.

Steven Bethard, Guergana Savova, Wei-Te Chen, Leon Derczynski, James Pustejovsky, and Marc Verhagen. 2016. SemEval-2016 task 12: Clinical TempEval. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 1052–1062, San Diego, CA, USA.

Steven Bethard, Guergana Savova, Martha Palmer, and James Pustejovsky. 2017. SemEval-2017 task 12: Clinical TempEval. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval 2017)*, pages 565–572, Vancouver, Canada.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS 2020)*, volume 33, pages 1877–1901.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with GPT-4. *arXiv preprint arXiv:2303.12712*.

Nathanael Chambers, Taylor Cassidy, Bill McDowell, and Steven Bethard. 2014. Dense event ordering with a multi-pass architecture. *Transactions of the Association for Computational Linguistics*, 2:273–284.

Yohan Bonescki Gumiel, Lucas Emanuel Silva e Oliveira, Vincent Claveau, Natalia Grabar, Emerson Cabrera Paraiso, Claudia Moro, and Deborah Ribeiro Carvalho. 2021. Temporal relation extraction in clinical texts: a systematic review. *ACM Computing Surveys (CSUR)*, 54(7):1–36.

Hasham Ul Haq, Veysel Kocaman, and David Talby. 2021. Deeper clinical document understanding using relation extraction. *arXiv preprint arXiv:2112.13259*.

Raghav Jain, Daivik Sojitra, Arkadeep Acharya, Sriparna Saha, Adam Jatowt, and Sandipan Dandapat. 2023. Do language models have a common sense regarding time? Revisiting temporal commonsense reasoning in the era of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6750–6774, Singapore.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Abdulrahman Khalifa, Sumithra Velupillai, and Stephane Meystre. 2016. UtahBMI at SemEval-2016 task 12: extracting temporal information from clinical text. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, pages 1256–1262, San Diego, CA, USA.

Timotej Knez and Slavko Žitnik. 2024. Multimodal learning for temporal relation extraction in clinical texts. *Journal of the American Medical Informatics Association*, 31(6):1380–1387.

Hee-Jin Lee, Hua Xu, Jingqi Wang, Yaoyun Zhang, Sungrim Moon, Jun Xu, and Yonghui Wu. 2016. UTHealth at SemEval-2016 task 12: an end-to-end system for temporal information extraction from clinical notes. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, pages 1292–1297, San Diego, CA, USA.

Chen Lin, Timothy Miller, Dmitriy Dligach, Steven Bethard, and Guergana Savova. 2019. A BERT-based universal model for both within-and cross-sentence clinical temporal relation extraction. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 65–71, Minneapolis, MN, USA.

Jian Liu, Jinan Xu, Yufeng Chen, and Yujie Zhang. 2021. Discourse-level event temporal ordering with uncertainty-guided graph completion. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI 2021)*, pages 3871–3877, Montreal, Canada.

Qiang Ning, Zhili Feng, and Dan Roth. 2017. A structured learning approach to temporal relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1027–1037, Copenhagen, Denmark.

Qiang Ning, Zhili Feng, Hao Wu, and Dan Roth. 2018. Joint reasoning for temporal and causal relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2278–2288, Melbourne, Australia.

William F Styler IV, Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet C De Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, et al. 2014. Temporal annotation in the clinical domain. *Transactions of the association for computational linguistics*, 2:143–154.

Weiyi Sun, Anna Rumshisky, and Ozlem Uzuner. 2013. Evaluating temporal relations in clinical text: 2012 i2b2 challenge. *Journal of the American Medical Informatics Association*, 20(5):806–813.

Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2023. Towards benchmarking and improving the temporal reasoning capability of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics Volume 1: Long Papers*, pages 14820–14835, Toronto, Canada.

Buzhou Tang, Yonghui Wu, Min Jiang, Yukun Chen, Joshua C Denny, and Hua Xu. 2013. A hybrid system for temporal information extraction from clinical text. *Journal of the American Medical Informatics Association*, 20(5):828–835.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on Gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Julien Tourille, Olivier Ferret, Aurélie Névéol, and Xavier Tannier. 2017. Neural architecture for temporal relation extraction: A Bi-LSTM approach for detecting narrative containers. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 224–230, Vancouver, Canada.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Hangyu Tu, Lifeng Han, and Goran Nenadic. 2023. Extraction of medication and temporal relation from clinical text using neural language models. In *2023 IEEE International Conference on Big Data (BigData)*, pages 2735–2744, Sorrento, Italy.

Sumithra Velupillai, Danielle L Mowery, Samir Abdelrahman, Lee Christensen, and Wendy Chapman. 2015. Blulab: Temporal information extraction for the 2015 clinical tempeval challenge. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 815–819, Denver, CO, USA.

Liang Wang, Peifeng Li, and Sheng Xu. 2022. DCT-Centered Temporal Relation Extraction. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2087–2097, Gyeongju, Republic of Korea.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022a. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022b. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837.

Georg Wenzel and Adam Jatowt. 2023. An overview of temporal commonsense reasoning and acquisition. *arXiv preprint arXiv:2308.00002*.

Chaoyi Wu, Weixiong Lin, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2023a. PMC-LLaMA: Towards Building Open-source Language Models for Medicine. *Preprint*, arXiv:2304.14454.

Zihao Wu, Lu Zhang, Chao Cao, Xiaowei Yu, Haixing Dai, Chong Ma, Zhengliang Liu, Lin Zhao, Gang Li, Wei Liu, et al. 2023b. Exploring the trade-offs: Unified large language models vs local fine-tuned models for highly-specific radiology NLI task. *arXiv preprint arXiv:2304.09138*.

Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata. 2019. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 41(09):2251–2265.

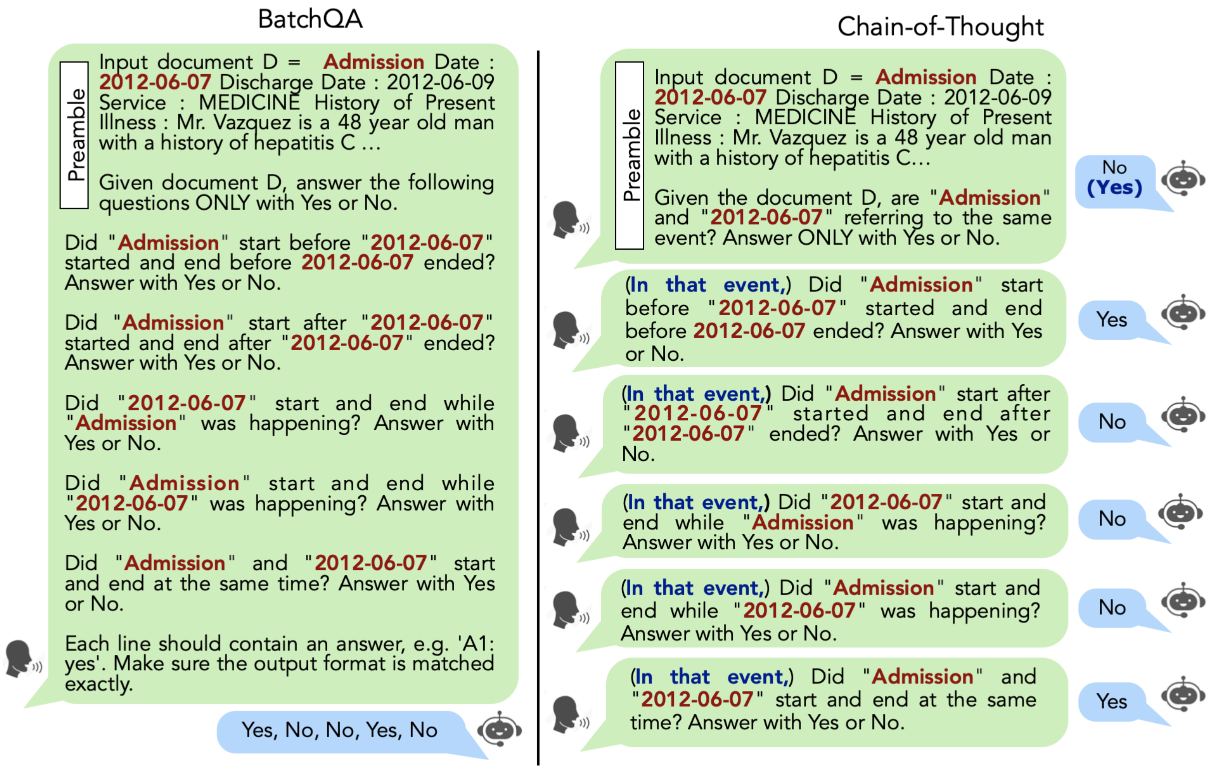
Hao-Ren Yao, Luke Breitfeller, Aakanksha Naik, Chunxiao Zhou, and Carolyn Rose. 2022. Multi-scale contrastive knowledge co-distillation for event temporal relation extraction. *arXiv preprint arXiv:2209.00568*.

Chenhan Yuan, Qianqian Xie, and Sophia Ananiadou. 2023. Zero-shot temporal relation extraction with ChatGPT. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 92–102, Toronto, Canada.

A Appendix

Technical details Getting responses from GPT-3.5 for all the pairs for both types of prompts costed around 800$ and lasted 27 hours. For the open-source models we used a single H100 GPU, and for the rest two H100 GPUs. The running time for each model was:

- Mixtral 8x7B BatchQA: 6 hours
- Mixtral 8x7B CoT: 48 hours
- Llama2 70B BatchQA: 24 hours
- Llama2 70B CoT: 7 days
- Gemma 7B BatchQA: 3 hours
- Gemma 7B CoT: 25 hours
- PMC-Llama 13B CoT: 2.5 days



Input document D = Admission Date : 2012-06-07 Discharge Date : 2012-06-09 Service : MEDICINE History of Present Illness : Mr. Vazquez is a 48 year old man with a history of hepatitis C ...

Given document D, answer the following questions ONLY with Yes or No.

Did "Admission" start before "2012-06-07" started and end before 2012-06-07 ended? Answer with Yes or No.

Did "Admission" start after "2012-06-07" started and end after "2012-06-07" ended? Answer with Yes or No.

Did "2012-06-07" start and end while "Admission" was happening? Answer with Yes or No.

Did "Admission" start and end while "2012-06-07" was happening? Answer with Yes or No.

Did "Admission" and "2012-06-07" start and end at the same time? Answer with Yes or No.

Each line should contain an answer, e.g. 'A1: yes'. Make sure the output format is matched exactly.

Yes, No, No, Yes, No

Input document D = Admission Date : 2012-06-07 Discharge Date : 2012-06-09 Service : MEDICINE History of Present Illness : Mr. Vazquez is a 48 year old man with a history of hepatitis C...

Given the document D, are "Admission" and "2012-06-07" referring to the same event? Answer ONLY with Yes or No.

No (Yes)

(In that event,) Did "Admission" start before "2012-06-07" started and end before 2012-06-07 ended? Answer with Yes or No.

Yes

(In that event,) Did "Admission" start after "2012-06-07" started and end after "2012-06-07" ended? Answer with Yes or No.

No

(In that event,) Did "2012-06-07" start and end while "Admission" was happening? Answer with Yes or No.

No

(In that event,) Did "Admission" start and end while "2012-06-07" was happening? Answer with Yes or No.

No

(In that event,) Did "Admission" and "2012-06-07" start and end at the same time? Answer with Yes or No.

Yes

Figure 4: Examples of an interaction with the LLM using two different prompting strategies: BatchQA and Chain-of-Thought.

r_1	r_2	$trans(r_1, r_2)$
before	before	before
after	after	after
includes	includes	includes
is included	is included	is included
simultaneous	simultaneous	simultaneous
before	simultaneous	before
after	simultaneous	after
includes	simultaneous	includes
is included	simultaneous	is included
before	after	[before, after, includes, is included, simultaneous]
before	includes	[before, includes]
before	is included	[before, is included]
after	before	[before, after, includes, is included, simultaneous]
after	includes	[after, includes]
after	is included	[after, is included]
includes	before	[before, includes]
includes	after	[after, includes]
includes	is included	[before, after, includes, is included, simultaneous]
is included	before	[before, is included]
is included	after	[after, is included]
is included	includes	[before, after, includes, is included, simultaneous]
simultaneous	before	before
simultaneous	after	after
simultaneous	includes	includes
simultaneous	is included	is included

Table 4: Transitivity rules for the five temporal relations used in this study.

r	$\bar{r}$
before	after
after	before
includes	is included
is included	includes
simultaneous	simultaneous

Table 5: Symmetry rules for the five temporal relations used in this study.

Overview of the First Shared Task on Clinical Text Generation: RRG24 and "Discharge Me!"

Justin Xu[*1] **Zhihong Chen**[*1] **Andrew Johnston**[1] **Louis Blankemeier**[1]

Maya Varma[1] **Jason Hom**[1] **William J. Collins**[1]

Ankit Modi[2] **Robert Lloyd**[2] **Benjamin Hopkins**[3]

Curtis P. Langlotz[1] **Jean-Benoit Delbrouck**[1]

[1]**Stanford University** [2]**University of Arizona**
[3]**University of Southern California**
{xujustin,zhihongc,drewj32,langlotz,jbdel}@stanford.edu

Abstract

Recent developments in natural language generation have tremendous implications for healthcare. For instance, state-of-the-art systems could automate the generation of sections in clinical reports to alleviate physician workload and streamline hospital documentation. To explore these applications, we present a shared task consisting of two subtasks: (1) Radiology Report Generation (RRG24) and (2) Discharge Summary Generation ("Discharge Me!"). RRG24 involves generating the 'Findings' and 'Impression' sections of radiology reports given chest X-rays. "Discharge Me!" involves generating the 'Brief Hospital Course' and 'Discharge Instructions' sections of discharge summaries for patients admitted through the emergency department. "Discharge Me!" submissions were subsequently reviewed by a team of clinicians. Both tasks emphasize the goal of reducing clinician burnout and repetitive workloads by generating documentation. We received 201 submissions from across 8 teams for RRG24, and 211 submissions from across 16 teams for "Discharge Me!".

1 Introduction

An important application of natural language generation (NLG) in medical artificial intelligence (AI) is radiology report generation (RRG). Specifically, an RRG system can be designed to accept radiology images (*e.g.,* chest X-rays) of a patient and generate a textual report describing the clinical observations in the images. This is a clinically important task, and offers the potential to reduce the repetitive work of radiologists and generally improve clinical communication (Pang et al., 2023). Existing studies have been conducted using a single dataset, which limits the scale and diversity of the data and results. Therefore, we introduce our first subtask, RRG24, where we curate Interpret-CXR, a large-scale collection of RRG datasets from a variety of different sources (*i.e.,* MIMIC-CXR (Johnson et al., 2019), CheXpert (Irvin et al., 2019), PadChest (Bustos et al., 2020), BIMCV-COVID19 (Vayá et al., 2020), and OpenI (Demner-Fushman et al., 2016)). In RRG24, participants generate the Findings and Impression sections from chest X-rays. We then evaluate the generations on common leaderboards with standard and recently proposed metrics. Ultimately, this task aims to benchmark recent progress using common data splits and evaluation implementations.

NLG can also impact discharge documentation by playing a role in generating discharge summaries. Hence, we introduce our second subtask, "Discharge Me!", with the primary objective of encouraging NLG systems that alleviate clinician burden when writing detailed discharge summaries. Clinicians play a crucial role in documenting patient progress after a hospital stay, but the creation of concise yet comprehensive Brief Hospital Course (BHC) sections and Discharge Instructions often demands a significant investment of time (Do et al., 2020; Alissa et al., 2021). These two sections in particular cannot be readily copied from prior notes, and thus must be written from scratch by clinicians who synthesize information from across the patient record (Weetman et al., 2021). This process

[*] Equal contribution

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 85–98
August 16, 2024. ©2024 Association for Computational Linguistics

contributes to clinician burnout and poses operational inefficiencies within hospital workflows (Haycock et al., 2014). We hypothesize that computer-generated clinical documentation has the potential to more accurately and completely capture a patient's hospital course while reducing the administrative burden on clinicians, which, in turn, mitigates burnout, streamlines hospital operations, and ultimately improves the quality of care. Thus, in "Discharge Me!", participants submit generations of both target sections (BHC & Discharge Instructions). We evaluate submissions on a common leaderboard and conduct a subsequent manual clinician review to measure clinical alignment of the outputs.

2 Related Work

2.1 Radiology Report Generation

Recent advances in computer vision (CV) and NLG have shown great potential for the automatic generation of radiology reports. This progress can be summarized from three perspectives:

- (1) Data: Most relevant studies focus on chest X-rays, mainly owing to the current number of publicly available image-report datasets for this modality (*e.g.,* MIMIC-CXR, PadChest, and OpenI, etc.). Recently, there have also been studies expanding the scope of radiology report generation to other modalities (*e.g.,* computed tomography (CT) (Loveymi et al., 2021; Hamamci et al., 2024) and ultrasound (Zeng et al., 2020; Yang et al., 2021; Huh et al., 2023)).
- (2) Methodology: The methods for radiology report generation have evolved from task-specific modeling to pre-training-based approaches. For the former, researchers have incorporated the task priors into the designs of the model architectures (Shin et al., 2016; Zhang et al., 2017; Jing et al., 2018; Chen et al., 2020; Zhang et al., 2020; Liu et al., 2021; Delbrouck et al., 2022a; Hou et al., 2023), whereas for the latter, researchers have performed domain-specific representation learning using vision encoders or have adopted large pre-trained language decoders (Thawkar et al., 2023; Hyland et al., 2023; Tu et al., 2024).
- (3) Evaluation: One of the largest factors hampering radiology report generation progress is the selection of evaluation metrics. Due to its domain-specific characteristics, simple n-gram matching metrics (*e.g.,* BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), METEOR (Banerjee and Lavie, 2005), and CIDEr (Vedantam et al., 2015)) are sub-optimal choices for this task. However, researchers have proposed various model-based metrics for evaluating the quality of generated reports, such as BERTScore (Zhang et al., 2019), F1-CheXbert (Smit et al., 2020), F1-RadGraph (Delbrouck et al., 2022a), and GREEN (Ostmeier et al., 2024).

2.2 Discharge Summary Generation

Previous research has also examined AI technologies for the generation of discharge summaries to alleviate clerical burden for clinicians. For instance, several studies investigated GPT-3.5's and GPT-4's capability to generate discharge notes in tandem with various prompting strategies. In a UK pilot feasibility study, it was observed that a set of 25 AI-generated summaries were all deemed acceptable by general practitioners, compared to 23/25 (92%) of summaries written by junior doctors (Clough et al., 2024). Other studies similarly concluded that these proprietary models exhibit great potential and are able to generate acceptable discharge summaries with minimal misinformation (Kim et al., 2024; Waisberg et al., 2023). However, despite being able to increase efficiency and reduce the time required for documentation as compared to writing or dictating notes, instances of hallucination or omission of clinically significant facts were observed for certain discharge summaries involving complex surgeries. As such, the factual correctness of these large language models (LLMs) for specific generation tasks could be improved (Williams et al., 2024; Dubinski et al., 2024).

Based on this, some studies have focused on generating a particular section common to most discharge summaries – BHC – optimizing for correctness and faithfulness. The BHC is a succinct summary of a patient's entire journey through the hospital and are embedded within complex discharge summaries. Efforts in compiling large-scale datasets for the generation of these BHC sections (Adams et al., 2021), including those with synthetic data (Adams et al., 2022), have led to subsequent contrastive learning methods for aligning generation models (Adams et al., 2023). Finally, methods leveraging heuristics to increase factuality (*e.g.,* retrieval and ontology referencing) have also been developed (Adams et al., 2024; Hartman et al., 2023).

Some research has similarly centered on the Discharge Instructions section, sometimes known as the Patient Instructions section. This section is patient-facing and details instructions for the patient to continue their care at home, such as information on diet, therapies, and medications, as well as any details for follow-up appointments. Patient readability of this section is critical, and LLMs could be used to reformulate them into a more patient-friendly language (Zaretsky et al., 2024). Similar to the BHC, previous work also explored frameworks for the generation of faithful Patient Instructions (Liu et al., 2022).

3 RRG24: Radiology Report Generation

RRG24 was hosted on ViLMedic (Delbrouck et al., 2022b), a modular framework for vision-language multimodal research in medicine. The library contains reference implementations for state-of-the-art vision-language architectures for medicine and also hosts shared challenges in AI. A total of 201 submissions were received from across 8 teams.

3.1 Data

We curated Interpret-CXR, a large-scale collection of RRG datasets from the following five sources: MIMIC-CXR (Johnson et al., 2019), CheXpert (Irvin et al., 2019), PadChest (Bustos et al., 2020), BIMCV-COVID19 (Vayá et al., 2020), and OpenI (Demner-Fushman et al., 2016). The breakdown of Interpret-CXR, including details of the four splits used in RRG24 (Training, Validation, Public Test, and Hidden Test) are reported in Table 1.

3.2 Evaluation

We applied two types of metrics to evaluate different systems: *n*-gram-based and model-based metrics. For the former, we adopted BLEU-4 (Papineni et al., 2002) and ROUGE-L (Lin, 2004), whereas for the latter, we adopted BERTScore (Zhang et al., 2019), F1-CheXbert (Smit et al., 2020), and F1-RadGraph (Delbrouck et al., 2022a). To standardize the evaluation process, we used the same script from ViLMedic to evaluate all systems. By doing so, we avoid different teams using different versions or hyperparameters for a given metric – for example, some existing studies use differing versions of BERTScore, leading to inconsistent score reporting.

3.3 Results

The automatic results for the Findings and Impression sections are shown in Tables 2 and 3, respectively (Note: *iHealth-Chile-1* did not submit scores for Impression generation, and thus is not included in Table 3). We congratulate *e-Health CSIRO*, *MAIRA*, and *AIRI* for their outstanding performance on both Findings and Impression generation. It is also worth highlighting that the other teams (*Gla-AI4BioMed*, *SICAR*, *CID*, *iHealth-Chile-3&2*, and *iHealth-Chile-1*) designed novel solutions as well, providing insights for future research in this field beyond the competition. We also ran an evaluation using GREEN for the top 2 best-scoring systems (*e-Health CSIRO* and *MAIRA*) and recorded scores of 36.9 and 35.2, respectively, aligning with the leaderboard rankings[1].

3.4 Descriptions of Systems

3.4.1 e-Health CSIRO

e-Health CSIRO (Nicolson et al., 2024) integrated entropy regularization into self-critical sequence training to help maintain a higher entropy in the token distribution, preventing overfitting to common phrases and ensuring a broader exploration of the vocabulary during training. They applied this to a multimodal language model with RadGraph as the reward. Additionally, their model incorporated several other features: (i) the use of type embeddings to differentiate between Findings and Impression section tokens; and (ii) the use of a non-causal attention mask for image embeddings and a causal mask for report token embeddings.

3.4.2 MAIRA

MAIRA (Srivastav et al., 2024) combined a CXR-specific image encoder with a pre-trained LLM (Vicuna-7B-v1.5) via a multi-layer perceptron (MLP) adapter of 4 layers. The image encoder is a ViT-B model that leverages DINOv2, a state-of-the-art self-supervised learning method. Both the LLM and the adapter are fine-tuned in a single stage training setup for RRG. Their results indicated that joint training for Findings and Impression prediction improves the metrics for Findings generation. Additionally, incorporating lateral images alongside frontal images further enhances all metrics. They showed that scaling the model size from Vicuna-7B to Vicuna-13B also improves metrics. To handle multiple predictions for a study (as each study can have multiple frontal and/or lateral images), they utilized GPT-4 to select the best report.

3.4.3 AIRI

AIRI (Samokhin et al., 2024) utilized the LLaVA framework, where the vision encoder is a DINOv2 trained on medical data and the language decoder is a specialized biomedical LLM. They used the same model to generate both Impressions and Findings with different prompts: "Write findings for this X-ray." or "Write impression for this X-ray.". The system prompt from LLaVA-Med (Li et al., 2024) was also used.

3.4.4 Gla-AI4BioMed

Gla-AI4BioMed (Zhang et al., 2024) leveraged the Vicuna-7B architecture and integrated a CLIP image encoder with a fine-tuned LLM. The model underwent a two-stage training process, whereby chest X-ray features are initially aligned with the language model, and said model is subsequently fine-tuned for report generation. The model processed multiple images simultaneously by stitching them together, mimicking the workflow of radiology professionals.

3.4.5 SICAR

SICAR (Udomlapsakul et al., 2024) incorporated the SigLIP vision encoder and the Phi-2-2.7B language model to train an efficient RRG model. They also implemented a novel two-stage post-processing pipeline. They first enhanced the readability and clarity of the reports, then cross-verified the model outputs by integrating X-Raydar, an advanced X-ray classification model, addressing false negatives.

3.4.6 CID

CID (Liao et al., 2024) proposed a novel paradigm for incorporating graph structural data into the RRG model. Their approach involved predicting graph labels based on visual features and subsequently initiating the decoding process through a template injection conditioned on the predicted labels. These results provided preliminary

[1] We adopted GREEN instead of the naive GPT-4 pairwise comparison since Ostmeier et al. (2024) found GPT-4 to have low correlation with expert preference.

Table 1: Dataset Breakdown of Interpret-CXR for RRG24

Dataset	Training		Validation		Public Test		Hidden Test	
	Findings	Impression	Findings	Impression	Findings	Impression	Findings	Impression
PadChest	101,752	-	1,112	4,589	-	-	-	-
BIMCV-COVID19	45,525	-	1,202	-	-	-	-	-
CheXpert	45,491	181,619	2,641	-	-	-	-	-
OpenI	3,252	3,628	85	92	-	-	-	-
MIMIC-CXR	148,374	181,166	3,799	4,650	-	-	-	-
Total	344,394	366,413	8,839	9,331	2,692	2,967	1,063	1,428

Table 2: RRG24 Leaderboard for the Findings Section

Rank	Team	Overall Score ↑	Automatic Evaluation Metrics ↑				
			BLEU-4	ROUGE-L	BERTScore	F1-CheXbert	F1-RadGraph
1	e-Health CSIRO	**35.56**	**11.68**	26.16	53.80	57.49	**28.67**
2	MAIRA	35.08	11.24	**26.58**	**54.22**	**57.87**	25.48
3	AIRI	33.55	9.97	25.82	52.42	54.25	25.29
4	Gla-AI4BioMed	31.01	7.65	24.35	52.69	46.21	24.13
5	SICAR	30.93	6.62	23.66	50.74	49.00	24.62
6	CID	30.71	7.46	23.30	50.89	50.47	21.45
7	iHealth-Chile-3&2	23.38	4.81	15.96	44.03	33.69	18.41
8	iHealth-Chile-1	20.83	6.46	20.51	49.23	9.35	18.59

evidence for the feasibility of this new approach, which warrants further exploration in the future.

3.5 iHealth-Chile-3&2

iHealth-Chile-3&2 (Loch et al., 2024) focused on exploring various template-based strategies using predictions from multi-label image classifiers as input, which was inspired by prior work on template-based report generation. Two approaches were explored: (i) a straightforward implementation from Pino et al. (2021) directly; and (ii) replacing the fully connected layer with an attention-based pooling mechanism conditioned on a fact embedding.

3.6 iHealth-Chile-1

iHealth-Chile-1 (Campanini et al., 2024) developed a new strategy for in-context learning. Their system is formed using a vision-encoder, a vision-language connector or adapter, and a LLM able to process text and visual embeddings. They also designed an enriched prompt by combining a standard instruction ("Write the finding section of a chest x-ray radiology report") with reports generated by a multi-label classifier and a group of template sentences.

3.7 Limitations & Challenges

The evaluation for medical text generation is challenging due to its domain-specific characteristics, making it difficult to measure performance as it relates to clinical utility. This challenge leveraged common metrics that are used by existing RRG studies. Unfortunately, these evaluations may be limited when considering the real-world clinical impact of the submitted systems.

4 "Discharge Me!": Discharge Summary Generation

"Discharge Me!" was hosted on Codabench (Xu et al., 2022), an open source platform used to organize various tasks and benchmarks. A total of 211 submissions was received from across 16 teams.

4.1 Data

Participants were provided a dataset derived from the MIMIC-IV-Note module (Johnson et al., 2023). The modified and filtered dataset included 109,168 hospital admissions from the Emergency Department (ED), split into four sets (Training, Validation, Phase I Test, and Phase II Test) (Xu, 2024). Each visit includes chief complaints and diagnosis codes (either ICD-9 or ICD-10) documented by the ED[2], at least one radiology report, and a discharge summary with both BHC and Discharge Instructions sections.

The generation targets for the BHC were extracted from the full discharge notes using a complex regular expression strategy that searched for relevant section headers and new-line formatting characters. A similar strategy was used for Discharge Instructions; however, given that this section is usually located at the end of a discharge note as its very last section, extraction was more trivial. Samples where the extracted length of either section was shorter than 10 words were removed

[2] We assume ED diagnosis codes are available to the discharging clinician as ED documentation is likely to be complete at the time of discharge in most cases. However, we acknowledge that ICD codes may not necessarily be finalized, so they will be removed in future iterations of the shared task.

Table 3: RRG24 Leaderboard for the Impression Section

Rank	Team	Overall Score ↑	Automatic Evaluation Metrics ↑				
			BLEU-4	ROUGE-L	BERTScore	F1-CheXbert	F1-RadGraph
1	e-Health CSIRO	**35.28**	**12.33**	28.32	50.94	**56.97**	**27.83**
2	MAIRA	34.06	11.66	**28.48**	**51.62**	53.27	25.26
3	AIRI	32.98	10.91	27.46	49.55	52.32	24.67
4	SICAR	30.73	8.03	24.29	47.15	52.73	21.46
5	Gla-AI4BioMed	30.46	9.60	25.27	48.60	46.74	22.10
6	CID	25.21	7.13	20.41	43.67	39.64	15.19
7	iHealth-Chile-3&2	17.30	1.66	10.21	37.21	25.82	11.58

and deemed invalid. The complete breakdown of the dataset is available in Table 4.

Participants were allowed to incorporate external datasets, either publicly available or proprietary, as well as link additional patient data from other MIMIC-IV modules. Additionally, with the exception of the test dataset, participants were given the flexibility of using all or part of the provided dataset in any combination as they see fit.

4.2 Evaluation

4.2.1 Automatic Scoring

Automatic scoring took place on Codabench with a Python 3.9 environment. A hidden subset of 250 samples from the test datasets of the respective phases was used to evaluate the submissions. The metrics for this task were based on a combination of textual similarity (*n*-gram-based lexical metrics) and factual correctness of the generated text. Specifically, we considered the following metrics to automatically score submissions: BLEU-4 (Papineni et al., 2002), ROUGE-1/-2/-L (Lin, 2004), BERTScore (Zhang et al., 2019), METEOR (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and MEDCON (Van Veen et al., 2024).

Initially, submissions were scored on both target sections separately (BHC & Discharge Instructions). The mean across all test samples were computed for each metric, resulting in several performance scores for each of the two target sections (not reported on the leaderboard). Then, for each metric, we took the mean of the scores for each of the two target sections (reported under the metric name on the leaderboard). Finally, we computed the mean once again over all the metrics to arrive at a final overall system score (reported as Overall Score on the leaderboard).

For instance, given N samples, suppose s is defined as the score for a given sample for a given metric, then the mean across all samples for a particular target section, S, would be calculated by:

$$S = \sum_{1}^{N}(s_i)/N \tag{1}$$

We then calculated β, the mean of a given metric over both target sections, for each of the 8 metrics using:

$$\beta = (S_{BHC} + S_{DischargeInstructions})/2 \tag{2}$$

Finally, the overall system score was calculated by taking the mean of the 8 β values:

$$Overall = \sum_{1}^{8}(\beta_i)/8 \tag{3}$$

4.2.2 Clinician Scoring

At the end of the competition, the submissions from the top 6 best-scoring teams were reviewed by a group of six clinicians with diverse experiences in a broad range of specialties (two adult hospitalists, two clinical informatics fellows trained in pediatrics, a neurosurgeon, and a radiologist). Generated sections were evaluated for their completeness, correctness, and readability, as well as in a holistic comparison against the reference target sections (ground truth). In particular, completeness evaluates whether the generated text captures the clinically important information available in the reference text. In cases where there is inaccurate information, correctness specifies whether and how likely this mistake would lead to unintended impacts in future care. Readability was only evaluated by the clinicians for the BHC section as the intended audience of the Discharge Instructions section is the patient. Finally, the holistic comparison aimed to capture overall clinician preference.

Clinicians were presented with the reference target sections and the generated target sections side-by-side on a web-based survey dashboard hosted via Streamlit. Additionally, the full discharge summary was available in case reviewers required further context. They were then presented with a series of multiple-choice questions capturing each of the above criteria in a scale from 1 to 5, where 1 was the most negative option, and 5 was the most positive option.

Each clinician was provided with generated samples from three teams for evaluation. To minimize recall bias, we presented the generated submissions from all three teams consecutively in a randomized order for one particular sample, before moving onto the next.

Each team's submission was evaluated by three separate clinician reviewers. Scores were averaged and several agreement and reliability scores were calculated, including Cohen's Kappa and Fleiss Kappa for inter-observer agreement (McHugh, 2012; Landis and Koch, 1977), as well as the intraclass correlation coefficient (ICC) (Liljequist et al., 2019).

Table 4: Dataset Breakdown for "Discharge Me!"

Item	Total Count	Training	Validation	Phase I Test	Phase II Test
Hospital Visits	109,168	68,785	14,719	14,702	10,962
Discharge Summaries	109,168	68,785	14,719	14,702	10,962
Radiology Reports	409,359	259,304	54,650	54,797	40,608
ED Stays & Chief Complaints	109,403	68,936	14,751	14,731	10,985
ED Diagnoses	218,376	138,112	29,086	29,414	21,764

4.3 Results

4.3.1 Automatic Evaluation

Automatic scoring of the submissions took place on Codabench's platform using queues connected to independent compute workers hosted on GCP. The final leaderboard on the Phase II Test set is available in Table 5.

A baseline performance was available for participants to benchmark their submissions. The baseline outputs were generated by a LLaMA-2-7B model finetuned on radiology reports from MIMIC-III (Johnson et al., 2016). While the system exhibited some clinical domain knowledge, it struggled due to the diverse formatting of discharge summaries, which greatly differed from that of the radiology reports in the training set. All submissions exceeded the baseline performance.

4.3.2 Clinician Evaluation

Overall clinician review scores are available in Table 6, and the specific rankings for the BHC and Discharge Instructions sections are shown in Tables 7 and 8, respectively (mean clinician scores are provided, along with their constituent scores in brackets). Interestingly, the rankings for the overall clinician review exactly reflected that of the automatic evaluation using the reported metrics.

Figure 4.3.2 illustrates the interobserver agreement between pairwise clinicians based on the Cohen's Kappa statistic calculated for common submissions reviewed. As not all clinicians reviewed the same subset of submissions, a statistic could not be calculated for all reviewers (*i.e.,* reviewer #5 and #6 did not have any submissions in common). There was rather poor agreement between most clinicians, likely due to subjective aspects of the evaluation and varying clinician preference during the holistic comparison.

However, the Fleiss Kappa value indicated that the reviews for the top 6 best-scoring submissions, where each submission was reviewed by 3 individual clinicians, exhibited substantial to almost perfect agreement (Table 6). Moderate reliability was also observed for the review methodology, as inferred from the presented range of ICC values.

4.3.3 Readability of Discharge Instructions

As the Discharge Instructions section is intended for patients who many not have medical training and knowledge of clinical acronyms, we decided to skip the clinician review and opted for an evaluation using common readability scores: the Flesch Reading Ease score and the Flesch–Kincaid Grade Level (Friedman and Hoffman-Goetz, 2006).

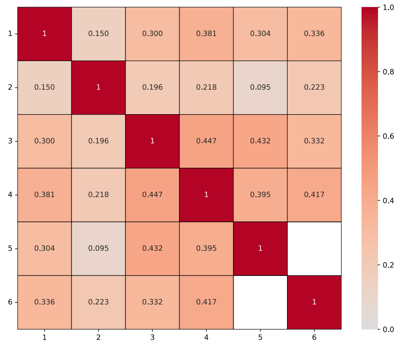


Figure 1: Correlation heatmap visualizing interobserver agreement between clinician reviews. Cohen's Kappa scores were computed between pairwise clinicians based on the respective common submission(s) reviewed.

The writing of patient-targeting notes at an appropriate readability level is crucial as it directly relates to patient comprehension, engagement, and adherence to treatment plans post-discharge. Several healthcare institutes have placed recommendations on the readability of patient-facing material. Specifically, the National Institutes of Health (NIH) and American Medical Association (AMA) encourage a reading grade level of not higher than sixth-grade, while the Centers for Disease Control and Prevention (CDC) suggests a reading grade level of lower than eighth-grade (Johnston et al., 2018; Cotugna et al., 2005; McCray, 2005; Burns et al., 2022).

A summary of the average readability metrics for the generated Discharge Instructions section is shown in Table 8. The readability of most submissions hovered around a reading grade level of seventh-grade, with the exception of one team at around the ninth-grade. The reference sections had a Flesch Reading Ease score of 61.81 (± 11.92) and a Flesch–Kincaid Grade Level of 8.16 (± 2.12). As such, all evaluated systems were able to reasonably re-create the readability of the reference sections, with several able to generate Discharge Instructions that are more understandable and in-line with established guidelines.

Table 5: “Discharge Me!” Automatic Scoring Leaderboard

Rank	Team	Overall Score ↑	Automatic Evaluation Metrics ↑							
			BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	METEOR	AlignScore	MEDCON
1	WisPerMed	**0.332**	**0.124**	**0.453**	**0.201**	**0.308**	**0.438**	**0.403**	**0.315**	**0.411**
2	HarmonAI Lab at Yale	0.300	0.106	0.423	0.180	0.284	0.412	0.381	0.265	0.353
3	aehrc	0.297	0.097	0.414	0.192	0.284	0.383	0.398	0.274	0.332
4	EPFL-MAKE	0.289	0.098	0.444	0.155	0.262	0.399	0.336	0.255	0.360
5	UF-HOBI	0.286	0.102	0.401	0.174	0.275	0.395	0.289	0.296	0.355
6	de ehren	0.284	0.097	0.404	0.166	0.265	0.389	0.376	0.231	0.339
7	DCT_PI	0.277	0.092	0.401	0.158	0.256	0.378	0.363	0.247	0.320
8	IgnitionInnovators	0.253	0.068	0.370	0.131	0.245	0.360	0.314	0.215	0.324
9	Shimo Lab	0.248	0.063	0.394	0.131	0.252	0.351	0.312	0.210	0.276
10	qub-cirdan	0.221	0.024	0.377	0.106	0.205	0.300	0.332	0.174	0.254
11	Roux-lette	0.206	0.030	0.319	0.084	0.182	0.289	0.287	0.195	0.265
12	UoG Siephers	0.191	0.017	0.341	0.109	0.209	0.268	0.247	0.143	0.193
13	mike-team	0.188	0.022	0.290	0.076	0.163	0.258	0.294	0.182	0.223
14	Ixa-UPV	0.183	0.016	0.259	0.057	0.144	0.282	0.284	0.210	0.215
15	MLBMIKABR	0.170	0.039	0.210	0.092	0.131	0.186	0.306	0.205	0.191
16	cyq	0.104	0.002	0.197	0.016	0.106	0.179	0.106	0.132	0.091

Table 6: “Discharge Me!” Clinician Scoring Leaderboard

Rank	Team	Average ↑	Fleiss Kappa	Intraclass Corr.
1	WisPerMed	**3.375**	0.781	0.336
2	HarmonAI Lab at Yale	2.903	0.944	0.656
3	aehrc	2.785	0.904	0.685
4	EPFL-MAKE	2.720	0.896	0.563
5	UF-HOBI	2.579	0.923	0.574
6	de ehren	2.335	0.908	0.740

4.4 Descriptions of Top Systems

A total of 12 system papers were received (Damm et al., 2024; Socrates et al., 2024; Wu et al., 2024; Lyu et al., 2024; He et al., 2024; Koontz et al., 2024; Guo et al., 2024; Liu et al., 2024; Frayling et al., 2024; Wendelken et al., 2024; Tang et al., 2024; Naskar et al., 2024). The top 6 best-scoring systems are detailed in this subsection.

4.4.1 WisPerMed

WisPerMed (Damm et al., 2024) investigated Dynamic Expert Selection (DES) consisting of a collection of LLMs fine-tuned and prompted for the task. They demonstrated that a DES system that chooses texts based on a specific length criteria performed the best on the given dataset. Thus, their objective with this strategy was to initially rank LLMs based on their archived overall scores. Subsequently, for each discharge summary, the generated sections (BHC & Discharge Instructions) from the best model that had a word count within the range of 100 to 180 words was selected. If no model generated a block of text with a word count within this range, the text with the minimum word count greater than 70 words was selected. In cases where no piece of text met these criteria (*i.e.,* shorter than 70 words), the text from the highest-ranked model was chosen. This approach emerged from the finding that longer pieces of medical text often led to hallucinations or repetitiveness.

4.4.2 HarmonAI Lab at Yale

The pipeline for **HarmonAI Lab at Yale** (Socrates et al., 2024) consisted of two BioBART-Large models. The one generating BHC sections was trained on all the preceding text prior to the BHC, while the Discharge Instructions model was trained on the BHC. The BHC model had an increased training dataset size due to shuffling and recombining the provided datasets. Default hyperparameter settings were largely used for training, with the exception of a lower learning rate. Models were trained for 2 epochs. For generation, a 4-beam search and limited repeats with an n-gram size of 3 was employed. The minimum output length was set to 200 tokens based on the word count summary statistics and, and the maximum output token length was restricted to 1024 tokens due to the model specifications.

4.4.3 aehrc

aehrc (Liu et al., 2024) used the content in the discharge summary note prior to the target sections as input context for both training and inference. To better handle the distinctions between the two sections, the team trained two separate models to generate the BHC and the Discharge Instructions. Their best model was based on PRIMERA, which is an encoder-decoder language model that is capable of handling extended input contexts and generating longer outputs. This model offered a slight edge over fine-tuning popular decoder-based LLMs at the 7/8B parameter-level with LoRA, and was also significantly faster at inference. Beam search with a size of 4 was used for decoding.

4.4.4 EPFL-MAKE

EPFL-MAKE (Wu et al., 2024) mainly focused on the full-text available in the dataset as they believed that most of the useful information is hidden within. The text was used as an input into their system, which first extracted all sections that contained clinically useful information. The system then combined them into a new input. Some sections may have been removed if the new input was deemed too lengthy. The pre-processed input was then put into the medical LLM Meditron-7B, which is currently one of the top open-source medically pre-trained LLMs at the 7B level, to generate the BHC and Discharge Instructions sections.

Table 7: “Discharge Me!” Rankings based on Clinician Scoring of the Brief Hospital Course Section

Rank	Team	Average ↑	Clinician Evaluation Criteria ↑			
			Completeness	Correctness	Readability	Holistic Comparison
1	WisPerMed	**3.29**	**3.67 (4.08 3.16 3.76)**	**3.67 (4.20 3.40 3.40)**	**3.37 (3.76 3.40 2.96)**	**2.44 (2.96 2.60 1.76)**
2	EPFL-MAKE	2.58	3.29 (3.28 3.20 3.40)	2.83 (2.80 2.96 2.72)	2.53 (2.88 2.56 2.16)	1.65 (2.12 1.52 1.32)
3	UF-HOBI	2.49	2.48 (2.52 2.48 2.44)	3.36 (3.48 3.28 3.32)	2.71 (3.20 2.96 1.96)	1.41 (1.96 1.20 1.08)
4	HarmonAI Lab at Yale	2.44	3.52 (3.32 3.64 3.60)	2.59 (2.68 3.00 2.08)	2.11 (2.36 2.00 1.96)	1.53 (1.60 1.84 1.16)
5	de ehren	2.27	2.28 (2.36 2.32 2.16)	2.99 (3.12 3.24 2.60)	2.68 (2.72 2.84 2.48)	1.12 (1.16 1.20 1.00)
6	aehrc	2.10	2.31 (2.24 2.52 2.16)	3.05 (3.32 3.40 2.44)	1.96 (2.16 1.80 1.92)	1.09 (1.08 1.20 1.00)

Table 8: “Discharge Me!” Rankings based on Clinician Scoring of the Discharge Instructions Section

Rank	Team	Average ↑	Clinician Evaluation Criteria ↑			Flesch	Flesch-Kincaid
			Completeness	Correctness	Holistic Comparison	Reading Ease	Grade Level
1	aehrc	**3.69**	3.91 (3.80 4.40 3.52)	**4.55 (4.52 4.48 4.64)**	**2.63 (2.48 3.24 2.16)**	62.05 (± 10.04)	7.80 (± 1.76)
2	HarmonAI Lab at Yale	3.52	**4.27 (3.88 4.40 4.52)**	3.95 (3.84 3.88 4.12)	2.36 (2.36 2.40 2.32)	61.14 (± 14.52)	8.60 (± 4.19)
3	WisPerMed	3.49	3.95 (4.36 3.36 4.12)	4.00 (4.36 3.60 4.04)	2.53 (2.48 2.76 2.36)	63.35 (± 8.827)	7.48 (± 1.53)
4	EPFL-MAKE	2.91	3.45 (3.28 3.36 3.72)	3.41 (3.36 3.20 3.68)	1.87 (2.20 1.64 1.76)	58.72 (± 10.67)	9.04 (± 1.81)
5	UF-HOBI	2.70	3.01 (2.60 3.24 3.20)	3.29 (3.36 3.28 3.24)	1.79 (2.00 1.84 1.52)	66.73 (± 10.23)	6.96 (± 1.57)
6	de ehren	2.43	2.81 (2.84 3.12 2.48)	3.05 (3.36 3.12 2.68)	1.41 (1.44 1.60 1.20)	65.76 (± 8.706)	7.28 (± 1.84)

4.4.5 UF-HOBI

In their system, **UF-HOBI** (Lyu et al., 2024) employed two clinical LLMs that they have developed in their previous works, including an encoder-based model GatorTron (Yang et al., 2022) and a decoder-based model GatorTronGPT (Peng et al., 2023). The team adopted GatorTron to extract clinical concepts from the discharge summary notes, and utilized GatorTronGPT to generate the BHC and Discharge Instructions sections. GatorTron, which was fine-tuned on the 2010 i2b2 Challenge Named Entity Recognition (NER) dataset, was used to extract three categories of concepts (“TEST”, “PROBLEM”, and “TREATMENT”) from the discharge summary and radiology reports for each visit. The extracted concepts were then used to form the generation model input. Two GatorTronGPT models were then trained using the P-tuning strategy for the generation of the two respective target sections. The model inputs were thus the concepts extracted from the various other sections.

4.4.6 de ehren

de ehren utilized Meerkat-7B-v1.0, a compact, instruction-tuned medical AI system renowned for its advanced medical reasoning capabilities. Meerkat excelled in various medical Question Answering (QA) benchmarks, notably achieving a score of 74.3 on MedQA. To further scrutinize its performance in long-form text generation and summarization tasks within the clinical domain, the team selectively extracted key sections from discharge summaries to fine-tune the model with regards to the model’s attention window size.

4.5 Limitations & Challenges

A primary concern was the risk of data leakage due to the release of the test sets with ground truth sections. To mitigate this, two test sets were released in two phases (one released at the start and one released much closer to the submission deadline), and the final evaluation was conducted on a hidden subset of 250 samples selected from the test datasets of the respective phases. This approach aimed to discourage participants from using the ground truth for model inference, or from optimizing systems for the tasks metrics throughout the entire duration of the competition. However, this method ultimately relies on the adherence of the participants to task guidelines.

The task also faced the challenge of dealing with inconsistently formatted free-text where ground truth generation targets are embedded within. The nature of clinical free-text can vary greatly, making it difficult to standardize inputs.

Furthermore, certain sections of the discharge summary appearing after the generation targets may not be reasonably available to the clinician at the time of discharge and the writing of the discharge summary. This presents a dilemma, as using such information would not accurately reflect the clinician’s workflow. Although teams were reminded to justify any decisions made regarding the use of discharge summary sections, it was challenging to moderate this aspect.

Another limitation was the need to select discharge summaries of a reasonable length to make clinician review feasible. This selection process may introduce a bias, as longer or more complex summaries that could benefit from automated generation might be excluded. There was also plausible comparison bias during clinician review as clinicians were asked to review submissions that could have varied greatly in quality. However, we aimed to reduce this by randomizing the order in which submissions were presented to the clinicians.

5 Conclusion

As seen from the scores of the participating models for both tasks, there is great complexity in generating coherent, accurate, and clinically relevant free-text reports. Several factors contribute to this, including the inherent variability and nuance of natural language used in clinical settings.

It may be worthwhile to consider alternative approaches for fully automated report generation, such as by pre-processing reports into structured formats prior to AI generation. By breaking down the report generation process into more manageable tasks, generation systems may be able to achieve higher accuracy and coherence in their outputs (Lederman et al., 2022). However, the standardization of formatting for these reports poses a significant challenge due to the diversity of writing styles and training among clinicians.

A previous study also explored the feasibility of generating hospital discharge summaries by tracing the source origin of medical expressions that make up the report (Ando et al., 2022). Interestingly, the analysis found that a significant portion of the discharge summary originates from external sources rather than inpatient records, such as past clinical records, referral notes, and the expertise of the writing clinician. This suggests that an end-to-end generation pipeline would depend on advanced data retrieval and may ultimately require some form of manual clinician oversight.

Ultimately, we hope that this challenge will bolster the efforts of the biomedical natural language processing community in developing effective solutions for clinical text generation. We believe this task could form a solid foundation for future work on generating entire radiology reports or discharge summaries, which would help significantly reduce the time clinicians spend on administrative tasks and improve patient care quality.

References

Griffin Adams, Emily Alsentzer, Mert Ketenci, Jason Zucker, and Noémie Elhadad. 2021. What's in a Summary? Laying the Groundwork for Advances in Hospital-Course Summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4794–4811, Online. Association for Computational Linguistics.

Griffin Adams, Bichlien Nguyen, Jake Smith, Yingce Xia, Shufang Xie, Anna Ostropolets, Budhaditya Deb, Yuan-Jyue Chen, Tristan Naumann, and Noémie Elhadad. 2023. What are the Desired Characteristics of Calibration Sets? Identifying Correlates on Long Form Scientific Summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10520–10542, Toronto, Canada. Association for Computational Linguistics.

Griffin Adams, Han-Chin Shing, Qing Sun, Christopher Winestock, Kathleen McKeown, and Noémie Elhadad. 2022. Learning to Revise References for Faithful Summarization. ArXiv:2204.10290 [cs].

Griffin Adams, Jason Zucker, and Noémie Elhadad. 2024. SPEER: Sentence-Level Planning of Long Clinical Summaries via Embedded Entity Retrieval. ArXiv:2401.02369 [cs].

Rana Alissa, Jennifer A. Hipp, and Kendall Webb. 2021. Saving Time for Patient Care by Optimizing Physician Note Templates: A Pilot Study. *Frontiers in Digital Health*, 3:772356.

Kenichiro Ando, Takashi Okumura, Mamoru Komachi, Hiromasa Horiguchi, and Yuji Matsumoto. 2022. Is artificial intelligence capable of generating hospital discharge summaries from inpatient records? *PLOS digital health*, 1(12):e0000158.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Shohei T. Burns, Nwamaka Amobi, Joshua Vic Chen, Meghan O'Brien, and Lawrence A. Haber. 2022. Readability of Patient Discharge Instructions. *Journal of General Internal Medicine*, 37(7):1797–1798.

Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria De La Iglesia-Vaya. 2020. Padchest: A large chest x-ray image dataset with multi-label annotated reports. *Medical image analysis*, 66:101797.

Diego Campanini, Oscar Loch, Pablo Messina, Rafael Elberg, and Denis Parra. 2024. ihealth-chile-1 at rrg24: In-context learning and finetuning of a large multimodal model for radiology report generation. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Zhihong Chen, Yan Song, Tsung-Hui Chang, and Xiang Wan. 2020. Generating radiology reports via memory-driven transformer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1439–1449.

Reece Alexander James Clough, William Anthony Sparkes, Oliver Thomas Clough, Joshua Thomas Sykes, Alexander Thomas Steventon, and Kate King. 2024. Transforming healthcare documentation: harnessing the potential of AI to generate discharge summaries. *BJGP open*, 8(1):BJGPO.2023.0116.

Nancy Cotugna, Connie E. Vickery, and Kara M. Carpenter-Haefele. 2005. Evaluation of literacy level of patient education pages in health-related journals. *Journal of Community Health*, 30(3):213–219.

Hendrik Damm, Tabea M. G. Pakull, Bahadır Eryılmaz, Helmut Becker, Ahmad Idrissi-Yaghir, Henning Schäfer, Sergej Schultenkämper, and Christoph M. Friedrich. 2024. Wispermed at "discharge me!": Advancing text generation in healthcare with large language models, dynamic expert selection, and priming techniques on mimic-iv. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Jean-Benoit Delbrouck, Pierre Chambon, Christian Bluethgen, Emily Tsai, Omar Almusa, and Curtis Langlotz. 2022a. Improving the factual correctness of radiology report generation with semantic rewards. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4348–4360.

Jean-benoit Delbrouck, Khaled Saab, Maya Varma, Sabri Eyuboglu, Pierre Chambon, Jared Dunnmon, Juan Zambrano, Akshay Chaudhari, and Curtis Langlotz. 2022b. Vilmedic: a framework for research at the intersection of vision and language in medical ai. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 23–34.

Dina Demner-Fushman, Marc D Kohli, Marc B Rosenman, Sonya E Shooshan, Laritza Rodriguez, Sameer Antani, George R Thoma, and Clement J McDonald. 2016. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association*, 23(2):304–310.

Huy M. Do, Lillian G. Spear, Moozhan Nikpanah, S. Mojdeh Mirmomen, Laura B. Machado, Alexandra P. Toscano, Baris Turkbey, Mohammad Hadi Bagheri, James L. Gulley, and Les R. Folio. 2020. Augmented Radiologist Workflow Improves Report Value and Saves Time: A Potential Model for Implementation of Artificial Intelligence. *Academic Radiology*, 27(1):96–105.

Daniel Dubinski, Sae-Yeon Won, Svorad Trnovec, Bedjan Behmanesh, Peter Baumgarten, Nazife Dinc, Juergen Konczalla, Alvin Chan, Joshua D. Bernstock, Thomas M. Freiman, and Florian Gessler. 2024. Leveraging artificial intelligence in neurosurgery-unveiling ChatGPT for neurosurgical discharge summaries and operative reports. *Acta Neurochirurgica*, 166(1):38.

Erlend Frayling, Jake Lever, and Graham McDonald. 2024. Uog siephers at discharge me!: Exploring ways to process multi-part electronic health records for sequence to sequence generation. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Daniela B. Friedman and Laurie Hoffman-Goetz. 2006. A systematic review of readability and comprehension instruments used for print and web-based cancer information. *Health Education & Behavior: The Official Publication of the Society for Public Health Education*, 33(3):352–373.

Rui Guo, Greg Farnan, Niall McLaughlin, and Barry Devereux. 2024. Qub-cirdan at "discharge me!": Zero shot discharge letter generation by open-source llm. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Ibrahim Ethem Hamamci, Sezgin Er, and Bjoern Menze. 2024. CT2Rep: Automated Radiology Report Generation for 3D Medical Imaging. ArXiv:2403.06801 [cs, eess] version: 1.

Vince C. Hartman, Sanika S. Bapat, Mark G. Weiner, Babak B. Navi, Evan T. Sholle, and Thomas R. Campion. 2023. A method to automate the discharge summary hospital course for neurology patients. *Journal of the American Medical Informatics Association: JAMIA*, 30(12):1995–2003.

Michael Haycock, Laura Stuttaford, Oliver Ruscombe-King, Zoe Barker, Kathryn Callaghan, and Timothy Davis. 2014. Improving the percentage of electronic discharge summaries completed within 24 hours of discharge. *BMJ Open Quality*, 3(1):u205963.w2604. Publisher: BMJ Open Quality Section: BMJ Quality Improvement Programme.

Yunzhen He, Hiroaki Yamagiwa, and Hidetoshi Shimodaira. 2024. Shimo lab at "discharge me!": Discharge summarization by prompt-driven concatenation of electronic health record sections. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Wenjun Hou, Kaishuai Xu, Yi Cheng, Wenjie Li, and Jiang Liu. 2023. Organ: Observation-guided radiology report generation via tree reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8108–8122.

Jaeyoung Huh, Hyun Jeong Park, and Jong Chul Ye. 2023. Breast Ultrasound Report Generation using LangChain. ArXiv:2312.03013 [cs, eess].

Stephanie L Hyland, Shruthi Bannur, Kenza Bouzid, Daniel C Castro, Mercy Ranjit, Anton Schwaighofer, Fernando Pérez-García, Valentina Salvatelli, Shaury Srivastav, Anja Thieme, et al. 2023. Maira-1: A specialised large multimodal model for radiology report generation. *arXiv preprint arXiv:2311.13668*.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. 2019. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597.

Baoyu Jing, Pengtao Xie, and Eric Xing. 2018. On the automatic generation of medical imaging reports. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2577–2586.

A Johnson, T Pollard, S Horng, LA Celi, and R Mark. 2023. Mimic-iv-note: Deidentified free-text clinical notes (version 2.2). physionet.

Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3(1):160035. Publisher: Nature Publishing Group.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317.

David Johnston, Owen McMurray, Michael McKee, Michael McConville, and Niall Leonard. 2018. 'DISCHARGE LETTER QUALITY; HOW TO HELP BOTH JUNIOR DOCTORS AND GPS?'. *The Ulster Medical Journal*, 87(2):130.

Hanjae Kim, Hee Min Jin, Yoon Bin Jung, and Seng Chan You. 2024. Patient-Friendly Discharge Summaries in Korea Based on ChatGPT: Software Development and Validation. *Journal of Korean Medical Science*, 39(16):e148.

Jordan Koontz, Maite Oronoz, and Alicia Pérez. 2024. Ixa-med at discharge me! retrieval-assisted generation for streamlining discharge documentation. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

J. R. Landis and G. G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Asher Lederman, Reeva Lederman, and Karin Verspoor. 2022. Tasks as needs: reframing the paradigm of clinical natural language processing research for real-world decision support. *Journal of the American Medical Informatics Association*, 29(10):1810–1817.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. 2024. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems*, 36.

Yuxiang Liao, Yuanbang Liang, Yipeng Qin, Hantao Liu, and Irena Spasić. 2024. Cid at rrg24: Attempting in a conditionally initiated decoding of radiology report generation with clinical entities. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

David Liljequist, Britt Elfving, and Kirsti Skavberg Roaldsen. 2019. Intraclass correlation – A discussion and demonstration of basic features. *PLoS ONE*, 14(7):e0219854.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Fenglin Liu, Xian Wu, Shen Ge, Wei Fan, and Yuexian Zou. 2021. Exploring and distilling posterior and prior knowledge for radiology report generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13753–13762.

Fenglin Liu, Bang Yang, Chenyu You, Xian Wu, Shen Ge, Zhangdaihong Liu, Xu Sun, Yang Yang, and David A. Clifton. 2022. Retrieve, Reason, and Refine: Generating Accurate and Faithful Patient Instructions. In *Proceedings of the Neural Information Processing Systems*.

Jinghui Liu, Aaron Nicolson, Jason Dowling, Bevan Koopman, and Anthony Nguyen. 2024. e-health csiro at "discharge me!" 2024: Generating discharge summary sections with fine-tuned language models. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Oscar Loch, Pablo Messina, Rafael Elberg, Diego Campanini, Álvaro Soto, René Vidal, and Denis Parra. 2024. ihealth-chile-3&2 at rrg24: Template based report generation. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Samira Loveymi, Mir Hossein Dezfoulian, and Muharram Mansoorizadeh. 2021. Automatic Generation of Structured Radiology Reports for Volumetric Computed Tomography Images Using Question-Specific Deep Feature Extraction and Learning. *Journal of Medical Signals and Sensors*, 11(3):194–207.

Mengxian Lyu, Cheng Peng, Daniel Paredes, Ziyi Chen, Aokun Chen, Jiang Bian, and Yonghui Wu. 2024. Uf-hobi at "discharge me!": A hybrid solution for discharge summary generation through prompt-based tuning of gatortrongpt models. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Alexa T. McCray. 2005. Promoting Health Literacy. *Journal of the American Medical Informatics Association : JAMIA*, 12(2):152–163.

Mary L. McHugh. 2012. Interrater reliability: the kappa statistic. *Biochemia Medica*, 22(3):276–282.

Abir Naskar, Jane Hocking, Patty Chondros, Douglas Boyle, and Mike Conway. 2024. Mlbmikabr at "discharge me!": Concept based clinical text description generation. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Aaron Nicolson, Jinghui Liu, Jason Dowling, Anthony Nguyen, and Bevan Koopman. 2024. e-health csiro at rrg24: Entropy-augmented self-critical sequence training for radiology report generation. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Sophie Ostmeier, Justin Xu, Zhihong Chen, Maya Varma, Louis Blankemeier, Christian Bluethgen, Arne Edward Michalson, Michael Moseley, Curtis Langlotz, Akshay S. Chaudhari, and Jean-Benoit Delbrouck. 2024. GREEN: Generative Radiology Report Evaluation and Error Notation. ArXiv:2405.03595 [cs].

Ting Pang, Peigao Li, and Lijie Zhao. 2023. A survey on automatic generation of medical imaging reports based on deep learning. *BioMedical Engineering OnLine*, 22(1):48.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Cheng Peng, Xi Yang, Aokun Chen, Kaleb E. Smith, Nima PourNejatian, Anthony B. Costa, Cheryl Martin, Mona G. Flores, Ying Zhang, Tanja Magoc, Gloria Lipori, Duane A. Mitchell, Naykky S. Ospina, Mustafa M. Ahmed, William R. Hogan, Elizabeth A. Shenkman, Yi Guo, Jiang Bian, and Yonghui Wu. 2023. A study of generative large language model for medical research and healthcare. *npj Digital Medicine*, 6(1):1–10. Publisher: Nature Publishing Group.

Pablo Pino, Denis Parra, Cecilia Besa, and Claudio Lagos. 2021. Clinically correct report generation from chest x-rays using templates. In *Machine Learning in Medical Imaging: 12th International Workshop, MLMI 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, September 27, 2021, Proceedings 12*, pages 654–663. Springer.

V. Samokhin, M. Munkhoeva, and D. Umerenkov. 2024. Airi at rrg24: Llava with specialised encoder and decoder. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Hoo-Chang Shin, Kirk Roberts, Le Lu, Dina Demner-Fushman, Jianhua Yao, and Ronald M Summers. 2016. Learning to read chest x-rays: Recurrent neural cascade model for automated image annotation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2497–2506.

Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Y Ng, and Matthew Lungren. 2020. Combining automatic labelers and expert annotations for accurate radiology report labeling using bert. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1500–1519.

Vimig Socrates, Thomas Huang, Xuguang Ai, Soraya Fereydooni, Qingyu Chen, R. Andrew Taylor, and David Chartash. 2024. Yale at “discharge me!”: Evaluating constrained generation of discharge summaries with unstructured and structured information. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Shaury Srivastav, Mercy Ranjit, Fernando Pérez-García, Kenza Bouzid, Shruthi Bannur, Daniel C. Castro, Anton Schwaighofer, Harshita Sharma, Maximilian Ilse, Valentina Salvatelli, Sam Bond-Taylor, Fabian Falck, Anja Thieme, Hannah Richardson, Matthew P. Lungren, Stephanie L. Hyland, and Javier Alvarez-Valle. 2024. Maira at rrg24: A specialised large multimodal model for radiology report generation. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

An Quang Tang, Xiuzhen Zhang, and Minh Ngoc Dinh. 2024. Ignitioninnovators at “discharge me!”: Chain-of-thought instruction finetuning large language models for discharge summaries. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Omkar Thawkar, Abdelrahman Shaker, Sahal Shaji Mullappilly, Hisham Cholakkal, Rao Muhammad Anwer, Salman Khan, Jorma Laaksonen, and Fahad Shahbaz Khan. 2023. Xraygpt: Chest radiographs summarization using medical vision-language models. *arXiv preprint arXiv:2306.07971*.

Tao Tu, Shekoofeh Azizi, Danny Driess, Mike Schaekermann, Mohamed Amin, Pi-Chuan Chang, Andrew Carroll, Charles Lau, Ryutaro Tanno, Ira Ktena, et al. 2024. Towards generalist biomedical ai. *NEJM AI*, 1(3):AIoa2300138.

Kiartnarin Udomlapsakul, Parinthapat Pengpun, Tossaporn Saengja, Kanyakorn Veerakanjana, Krittamate Tiankanon, Pitikorn Khlaisamniang, Pasit Supholkhan, Amrest Chinkamol, Pubordee Aussavavirojekul, Hirunkul Phimsiri, Tara Sripo, Chiraphat Boonnag, Trongtum Tongdee, Thanongchai Siriapisith, Pairash Saiviroonporn, Jiramet Kinchagawat, and Piyalitt Ittichaiwong. 2024. Sicar at rrg2024: Gpu poor’s guide to radiology report generation. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, William Collins, Neera Ahuja, Curtis P. Langlotz, Jason Hom, Sergios Gatidis, John Pauly, and Akshay S. Chaudhari. 2024. Adapted large language models can outperform medical experts in clinical text summarization. *Nature Medicine*.

Maria De La Iglesia Vayá, Jose Manuel Saborit, Joaquim Angel Montell, Antonio Pertusa, Aurelia Bustos, Miguel Cazorla, Joaquin Galant, Xavier Barber, Domingo Orozco-Beltrán, Francisco García-García, et al. 2020. Bimcv covid-19+: a large annotated dataset of rx and ct images from covid-19 patients. *arXiv preprint arXiv:2006.01174*.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. CIDEr: Consensus-based Image Description Evaluation. ArXiv:1411.5726 [cs].

Ethan Waisberg, Joshua Ong, Mouayad Masalkhi, Sharif Amit Kamran, Nasif Zaman, Prithul Sarker, Andrew G. Lee, and Alireza Tavakkoli. 2023. GPT-4: a new era of artificial intelligence in medicine. *Irish Journal of Medical Science*, 192(6):3197–3200.

Katharine Weetman, Rachel Spencer, Jeremy Dale, Emma Scott, and Stephanie Schnurr. 2021. What makes a "successful" or "unsuccessful" discharge letter? Hospital clinician and General Practitioner assessments of the quality of discharge letters. *BMC Health Services Research*, 21:349.

S. Wendelken, A. Antony, R. Korutla, B. Pachipala, J. Shanahan, and W. Saba. 2024. Roux-lette at "discharge me!": Reducing ehr chart burden with a simple, scalable, clinician-driven ai approach. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Christopher Y. K. Williams, Jaskaran Bains, Tianyu Tang, Kishan Patel, Alexa N. Lucas, Fiona Chen, Brenda Y. Miao, Atul J. Butte, and Aaron E. Kornblith. 2024. Evaluating Large Language Models for Drafting Emergency Department Discharge Summaries. *medRxiv: The Preprint Server for Health Sciences*, page 2024.04.03.24305088.

Haotian Wu, Paul Boulenger, Antonin Faure, Berta Céspedes, Farouk Boukil, Nastasia Morel, Zeming Chen, and Antoine Bosselut. 2024. Epfl-make at "discharge me!": An llm system for automatically generating discharge summaries of clinical electronic health record. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Justin Xu. 2024. Discharge Me: BioNLP ACL'24 Shared Task on Streamlining Discharge Documentation.

Zhen Xu, Sergio Escalera, Adrien Pavão, Magali Richard, Wei-Wei Tu, Quanming Yao, Huan Zhao, and Isabelle Guyon. 2022. Codabench: Flexible, easy-to-use, and reproducible meta-benchmark platform. *Patterns*, 3(7):100543.

Shaokang Yang, Jianwei Niu, Jiyan Wu, Yong Wang, Xuefeng Liu, and Qingfeng Li. 2021. Automatic ultrasound image report generation with adaptive multimodal attention mechanism. *Neurocomputing*, 427:40–49.

Xi Yang, Aokun Chen, Nima PourNejatian, Hoo Chang Shin, Kaleb E. Smith, Christopher Parisien, Colin Compas, Cheryl Martin, Anthony B. Costa, Mona G. Flores, Ying Zhang, Tanja Magoc, Christopher A. Harle, Gloria Lipori, Duane A. Mitchell, William R. Hogan, Elizabeth A. Shenkman, Jiang Bian, and Yonghui Wu. 2022. A large language model for electronic health records. *npj Digital Medicine*, 5(1):1–9. Publisher: Nature Publishing Group.

Jonah Zaretsky, Jeong Min Kim, Samuel Baskharoun, Yunan Zhao, Jonathan Austrian, Yindalon Aphinyanaphongs, Ravi Gupta, Saul B. Blecker, and Jonah Feldman. 2024. Generative Artificial Intelligence to Transform Inpatient Discharge Summaries to Patient-Friendly Language and Format. *JAMA Network Open*, 7(3):e240357.

Xianhua Zeng, Li Wen, Banggui Liu, and Xiaojun Qi. 2020. Deep learning for ultrasound image caption generation based on object detection. *Neurocomputing*, 392:132–141.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Xi Zhang, Zaiqiao Meng, Jake Lever, and Edmond S.L. Ho. 2024. Gla-ai4biomed at rrg24: Visual instruction-tuned adaptation for radiology report generation. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Yixiao Zhang, Xiaosong Wang, Ziyue Xu, Qihang Yu, Alan Yuille, and Daguang Xu. 2020. When radiology report generation meets knowledge graph. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 12910–12917.

Zizhao Zhang, Yuanpu Xie, Fuyong Xing, Mason McGough, and Lin Yang. 2017. Mdnet: A semantically and visually interpretable medical image diagnosis network. In *Proceedings of the IEEE conference* *on computer vision and pattern recognition*, pages 6428–6436.

e-Health CSIRO at RRG24: Entropy-Augmented Self-Critical Sequence Training for Radiology Report Generation

Aaron Nicolson, Jinghui Liu, Jason Dowling, Anthony Nguyen, & Bevan Koopman
Australian e-Health Research Centre, CSIRO Health and Biosecurity, Brisbane, Australia
aaron.nicolson@csiro.au

Abstract

The Shared Task on Large-Scale Radiology Report Generation (RRG24) aims to expedite the development of assistive systems for interpreting and reporting on chest X-ray (CXR) images. This task challenges participants to develop models that generate the *findings* and *impression* sections of radiology reports from CXRs from a patient's study, using five different datasets. This paper outlines the e-Health CSIRO team's approach, which achieved multiple first-place finishes in RRG24. The core novelty of our approach lies in the addition of entropy regularisation to self-critical sequence training, to maintain a higher entropy in the token distribution. This prevents overfitting to common phrases and ensures a broader exploration of the vocabulary during training, essential for handling the diversity of the radiology reports in the RRG24 datasets. Our model is available on Hugging Face (https://huggingface.co/aehrc/cxrmate-rrg24).

1 Introduction

Machine learning holds the potential to significantly enhance diagnostic processes and clinical reporting, particularly within the field of radiology — a discipline characterised by high volumes of imaging data. Radiologists are often tasked with interpreting and reporting on hundreds of imaging studies daily, a repetitive process that is susceptible to fatigue and error. Automated systems capable of generating radiology reports from chest X-rays (CXRs) could greatly alleviate this burden by ensuring consistency and potentially reducing diagnostic turnaround times.

The Shared Task on Large-Scale Radiology Report Generation (RRG24) challenges participants to develop automated systems for producing textual reports from CXR images, with a particular focus on the findings and impression sections (Xu et al., 2024; Delbrouck et al., 2022b). These sections are crucial as they convey the diagnostic interpretation and clinical significance of a patient's study. The challenge provides a means to benchmark the various models under uniform conditions, offering insights into which approaches are most effective for CXR report generation. Participants were to train and evaluate their submissions on a dataset formed from five different sources, including MIMIC-CXR (Johnson et al., 2019), CheXpert (Chambon et al., 2024), PadChest (Bustos et al., 2020), BIMCV COVID-19 (Vayá et al., 2020), and Open-i IU X-ray (Demner-Fushman et al., 2016). This dataset consisted of four subsets, including the *training*, *validation*, *public-test*, and *hidden-test*, where the radiology reports were available for all except the hidden-test set. Finally, RRG24 presents participants with unique challenges to overcome, such as handling studies with missing sections and deciding whether to use a single model or separate models for each section.

This paper outlines the approach taken by team e-Health CSIRO in the RRG24 challenge. For this, we developed a multimodal language model that conditions report generation not only on previously generated words (or subwords), but also on the image embeddings of all the CXRs of a patient's study. We utilised a single model to generate both sections and incorporated special tokens to signify the absence of a section during training. These special tokens were also used to guide the model to generate specific sections during testing.

A key factor to the performance of our submissions was our modification to the self-critical sequence training (SCST) reinforcement learning (RL) algorithm (Rennie et al., 2017). A widely-used technique to enhance RL is to add entropy regularisation into the objective function. This approach boosts exploration and prevents the model from prematurely settling on less optimal actions (Mnih et al., 2016). Hence, we add entropy regularisation to SCST, forming Entropy-Augmented Self-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 99–104
August 16, 2024. ©2024 Association for Computational Linguistics

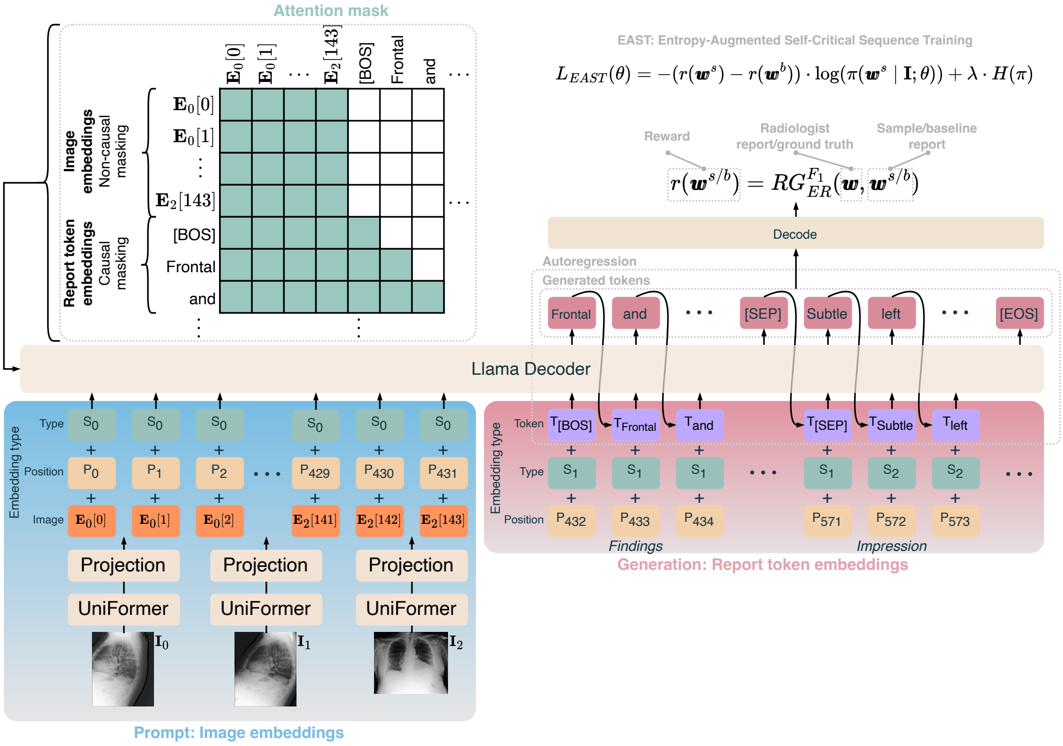

Attention mask
Image embeddings Non-causal masking
Report token embeddings Causal masking
E0[0]
E0[1]
E2[143]
[BOS]
Frontal
and
EAST: Entropy-Augmented Self-Critical Sequence Training
$L_{EAST}(\theta) = -(r(\boldsymbol{w}^s) - r(\boldsymbol{w}^b)) \cdot \log(\pi(\boldsymbol{w}^s \mid \mathbf{I}; \theta)) + \lambda \cdot H(\pi)$
Reward
Radiologist report/ground truth
Sample/baseline report
$r(\boldsymbol{w}^{s/b}) = RG^{F_1}_{ER}(\boldsymbol{w}, \boldsymbol{w}^{s/b})$
Decode
Autoregression
Generated tokens
Frontal
and
[SEP]
Subtle
left
[EOS]
Llama Decoder
Embedding type
Type
Position
Image
S0
P0
P1
P2
P429
P430
P431
E0[0]
E0[1]
E0[2]
E2[141]
E2[142]
E2[143]
Projection
UniFormer
I0
I1
I2
Token
T[BOS]
TFrontal
Tand
T[SEP]
TSubtle
Tleft
S1
S2
P432
P433
P434
P571
P572
P573
Findings
Impression
Generation: Report token embeddings
Prompt: Image embeddings

Figure 1: e-Health CSIRO's submission into RRG24, named CXRMate-RRG24. [BOS] denotes the *beginning-of-sentence* special token, [SEP] denotes the *separator* special token, and [EOS] denotes the *end-of-sentence* special token. $\mathbf{E}_k[i]$ is the i^{th} output of the projected last hidden state of the encoder for the k^{th} image of the study.

critical sequence Training (EAST). Using EAST, we optimised our model with RadGraph as the reward (Delbrouck et al., 2022a). RadGraph is the primary metric for RRG24; it evaluates the accuracy of a generated report by assessing how well the identified entities and their relationships align with those in a radiologist report. By optimising for this reward, we achieved multiple first-place finishes in RRG24.

2 Methodology

2.1 EAST: Entropy-Augmented Self-critical sequence Training

Entropy-Augmented Self-critical sequence Training (EAST) builds upon self-critical sequence training (SCST) by incorporating entropy regularisation. This encourages the model to maintain a higher entropy in its token distribution, thereby promoting diversity in token selection and preventing premature convergence on a smaller, selective set of tokens. The loss for SCST is as follows:

$$L_{SCST}(\theta) = -(r(\boldsymbol{w}^s) - r(\boldsymbol{w}^b)) \cdot \log(\pi(\boldsymbol{w}^s \mid \mathbf{I}; \theta)), \tag{1}$$

where $r(\boldsymbol{w}^s)$ is the reward for the sampled report ($\boldsymbol{w}^s = (w^s_1, ..., w^s_M)$ denotes the tokens of length M of the sampled report), $r(\boldsymbol{w}^b)$ is the reward for the baseline report ($\boldsymbol{w}^b = (w^b_1, ..., w^b_N)$ denotes the tokens of length N of the baseline report, where the baseline is generated with greedy search), $\mathbf{I} = [I_1, I_2, \ldots, I_K]$ denotes the images of a study (where K is the number of images in the study), θ represents the parameters of the model, and $\pi(\boldsymbol{w}^s \mid \mathbf{I}; \theta)$ denotes the policy under which $\boldsymbol{w}^s$ is sampled from. As illustrated in Figure 1, we utilise the RadGraph ER F1-score as the reward (Delbrouck et al., 2022a), where the generated report is either the sample or baseline report, both of which are compared to the radiologist report.

EAST is formed by adding an entropy term to $L_{SCST}(\theta)$:

$$L_{EAST}(\theta) = L_{SCST}(\theta) + \lambda \cdot H(\pi) \tag{2}$$

where λ is a coefficient that determines the weight of the entropy term in the loss function. The entropy is as follows:

$$H(\pi) = -\sum_{v \in \mathcal{V}} \pi(v \mid x; \theta) \log \pi(v \mid x; \theta), \tag{3}$$

Table 1: **Public test set** scores for the findings and impression sections (**presented as findings/impression**). The order of the leaderboard for RRG24 was determined by RadGraph-F1. The best scores are indicated in boldface.

Team/Method	BLEU-4	ROUGE-L	BERTScore	CheXbert-F1	RadGraph-F1
			e-Health CSIRO		
EAST	12.00/**9.43**	26.51/26.58	54.64/47.81	59.18/**57.73**	29.46/**27.01**
SCST	10.70/8.51	26.54/26.30	54.79/48.25	56.42/55.00	27.66/25.04
TF	11.63/7.52	25.92/23.34	51.34/41.46	50.73/47.27	23.12/20.08
			Top three teams besides ours		
`tartan`	**21.59**/-	**42.03**/-	**64.34**/-	**59.70**/-	**38.05**/-
`maira`	12.26/8.68	28.00/**28.40**	55.76/**50.48**	59.71/56.46	26.33/25.89
`airi`	10.13/7.10	26.54/25.92	53.84/47.18	55.49/51.33	25.82/24.07

where x represents the current state (as determined by the image embeddings and the previously generated tokens) and v represents a token from the vocabulary $\mathcal{V}$. This discourages the policy from converging too quickly to deterministic actions, thus encouraging the exploration of a wider set of generated reports.

2.2 Special Tokens and Missing Sections

As illustrated in Figure 1, our model generates both sections. To delineate these sections within the generated text, we utilise a separator token, following CXRMate (Nicolson et al., 2024a).[1] To accommodate reports during training that have a missing section, we employ two special tokens: `[NF]` for 'no findings' section and `[NI]` for 'no impression' section. They are used in place of the missing sections. They also facilitate the generation of specific sections as needed. For example, if only the impression section is to be generated, `[BOS][NF][SEP]` can be fed to the decoder to signal that the findings section is not to be generated. Furthermore, to encourage the generation of the impression section, the probability of the `[NI]` token can be set to zero.

2.3 Model

Our model, CXRMate-RRG24, is an evolution of our previous model, CXRMate, and is illustrated in Figure 1. We utilised UniFormer as the encoder (in particular, the 384×384 base model warm started with its token labelling fine-tuned checkpoint) (Li et al., 2023), which, in preliminary testing, performed comparably to the convolutional vision Transformer (CvT) (which we found to be the best performing encoder for CXR report generation in our previous work (Nicolson et al., 2023)) but significantly reduced the training time. The image embedding prompt is formed by processing each image in the study separately with the encoder and then projecting the encoder's last hidden state to match the decoder's hidden size using a learnable weight matrix. Each image was resized using bilinear interpolation so that its smallest side had a length of 384 and its largest side maintained the aspect ratio. Next, the resized image was cropped to a size of $\mathbb{R}^{3\times384\times384}$. The crop location was random during training and centred during testing. Following (Elgendi et al., 2021), the image was rotated around its centre during training, where the angle of rotation was sampled from $\mathcal{U}[-5°, 5°]$. Finally, the image was standardised using the statistics provided with the UniFormer checkpoint. A maximum of five images per study were used during training. If more were available, five were randomly sampled uniformly without replacement from the study.

For the decoder, we employed the Llama architecture, which is notable for features such as its rotary positional encoding (RoPE), root mean square normalisation (RMSNorm), and SwiGLU activation function (Touvron et al., 2023). The decoder was initialised randomly and used the CXRMate vocabulary, which was derived from the MIMIC-CXR training set. The hyperparameters of the Llama decoder mirror that of the CXRMate decoder, with six hidden layers, a hidden size of 768, 12 attention heads per layer, and an intermediate size of 3 072. Following CXRMate, we added source type embeddings to the input of the decoder to differentiate between findings and impression section tokens, as well as image embeddings. The max number of position embeddings was set to 2048 to accommodate both the image embeddings and the report token embeddings. The maximum number of tokens that could be generated was set to 512, which was also the limit for the radiologist reports during training. During testing, a beam size of four was utilised. Another factor that led to the use of the Llama decoder was the ease of providing a cus-

[1] `https://huggingface.co/aehrc/cxrmate`

Table 2: **Hidden test set** scores for the findings and impression sections (**presented as findings/impression**). The order of the leaderboard for RRG24 was determined by RadGraph-F1. The best scores are indicated in boldface.

Team/Method	BLEU-4	ROUGE-L	BERTScore	CheXbert-F1	RadGraph-F1
e-Health CSIRO					
EAST	**11.68/12.33**	26.16/28.32	53.80/50.94	57.49/**56.97**	**28.67/27.83**
SCST	10.25/10.95	26.10/27.34	53.88/50.07	55.78/54.79	27.29/24.97
TF	11.12/9.89	25.43/24.94	51.10/42.49	50.02/47.24	22.99/21.27
Top three teams besides ours					
`maira`	11.24/11.66	**26.58/28.48**	**54.22/51.62**	**57.87**/53.27	25.48/25.26
`airi`	9.97/10.91	25.82/27.46	52.42/49.55	54.25/52.32	25.29/24.67
`gla-ai4biomedic`	7.65/9.60	24.35/25.27	52.69/48.60	46.21/46.74	24.13/22.10

tom attention mask to current implementations.[2] This enabled non-causal masking to be utilised for the prompt and causal masking for the report token embeddings, as shown in Figure 1. This ensured that the self-attention heads were able to attend to all of the image embeddings at each position.

2.4 Training

Two stages of training were performed; teacher forcing (TF) (Williams and Zipser, 1989), followed by RL (either EAST or SCST). *AdamW* (Loshchilov and Hutter, 2022) was used for mini-batch gradient descent optimisation with an initial learning rate of 5e-5 for TF and 5e-6 for RL, a mini-batch size of 16 for TF and 8 for RL, a maximum of 32 epochs for TF and 1 epoch for RL, executed on a 94GB NVIDIA H100 GPU with FP32. For RL, validation was performed every $\frac{1}{50}$ of an epoch. The validation macro-averaged CheXbert F1 was the monitored metric for checkpoint selection. For RL, the sample report was generated with top-k sampling ($k = 50$). During RL, the encoder was frozen. For EAST, the entropy weight (λ) was set to 0.05.

3 Results and Discussion

The results for our key submissions on the public and hidden test sets are shown in Tables 1 and 2, respectively. The metrics utilised for RRG24 include BLEU-4 (Papineni et al., 2001), ROUGE-L (Lin and Och, 2004), BERTScore (Zhang et al., 2020), CheXbert-F1 (Smit et al., 2020), and RadGraph-F1 (Delbrouck et al., 2022a), the later of which is the primary metric used to rank the teams. Here, we compare TF, to SCST, and to our proposed method, EAST. EAST attained a higher score than TF for each metric, something SCST was not able to do (TF attained a higher BLEU-4 score than SCST for the findings section of both test datasets).

Comparing EAST to SCST, SCST attained a higher ROUGE-L score on the public-test findings sections, and a higher BERTScore on the public-test findings and impression sections, as well as the hidden-test findings sections. For all other cases, EAST demonstrated an improvement over SCST. Policies trained with entropy regularisation often have improved generalisation, as they have learnt to consider a broader set of possible actions. This may have led EAST to be more robust to the differing characteristics of each of the datasets used in the public and hidden test sets. With EAST, team e-Health CSIRO achieved a first-place finish amongst participants for the public-test impression sections and the hidden-test findings and impression sections. We also also achieved a second-place finish for the public-test findings sections. For a comparison of CXRMate-RRG24 to state-of-the-art methods in the literature, please see Nicolson et al. (2024b).

3.1 Conclusion

Our proposed approach, EAST, was able to generate reports that were quantitatively more aligned with radiologist reports than those generated using SCST. By incorporating entropy regularisation, EAST is able to maintain a higher diversity in token selection and mitigate overfitting to maintain generalisability. This was likely crucial in handling the varied characteristics of the datasets used in RRG24. While EAST shows promise, a more thorough investigation is required to validate its potential, including the impact of varying the entropy coefficient.

References

Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria De La Iglesia-Vaya. 2020. Padchest: A

[2] https://huggingface.co/blog/poedator/4d-masks

large chest x-ray image dataset with multi-label annotated reports. *Medical image analysis*, 66:101797.

Pierre Chambon, Jean-Benoit Delbrouck, Thomas Sounack, Shih-Cheng Huang, Zhihong Chen, Maya Varma, Steven QH Truong, Chu The Chuong, and Curtis P. Langlotz. 2024. CheXpert Plus: Augmenting a Large Chest X-ray Dataset with Text Radiology Reports, Patient Demographics and Additional Image Formats.

Jean-Benoit Delbrouck, Pierre Chambon, Christian Bluethgen, Emily Tsai, Omar Almusa, and Curtis Langlotz. 2022a. Improving the factual correctness of radiology report generation with semantic rewards. In *Findings of the Association for Computational Linguistics: EMNLP*, pages 4348–4360.

Jean-Benoit Delbrouck, Khaled Saab, Maya Varma, Sabri Eyuboglu, Pierre Chambon, Jared Dunnmon, Juan Zambrano, Akshay Chaudhari, and Curtis Langlotz. 2022b. ViLMedic: a framework for research at the intersection of vision and language in medical AI. In *ACL: System Demonstrations*, pages 23–34.

Dina Demner-Fushman, Marc D. Kohli, Marc B. Rosenman, Sonya E. Shooshan, Laritza Rodriguez, Sameer Antani, George R. Thoma, and Clement J. McDonald. 2016. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association*, 23(2):304–310.

Mohamed Elgendi, Muhammad Umer Nasir, Qunfeng Tang, David Smith, John-Paul Grenier, Catherine Batte, Bradley Spieler, William Donald Leslie, Carlo Menon, Richard Ribbon Fletcher, Newton Howard, Rabab Ward, William Parker, and Savvas Nicolaou. 2021. The Effectiveness of Image Augmentation in Deep Learning Networks for Detecting COVID-19: A Geometric Transformation Perspective. *Frontiers in Medicine*, 8.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. MIMIC-CXR, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317.

Kunchang Li, Yali Wang, Junhao Zhang, Peng Gao, Guanglu Song, Yu Liu, Hongsheng Li, and Yu Qiao. 2023. UniFormer: Unifying Convolution and Self-Attention for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10):12581–12600.

Chin-Yew Lin and Franz Josef Och. 2004. Automatic Evaluation of Machine Translation Quality Using Longest Common Subsequence and Skip-Bigram Statistics. In *ACL*, pages 605–612.

Ilya Loshchilov and Frank Hutter. 2022. Decoupled Weight Decay Regularization. In *ICLR*.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous Methods for Deep Reinforcement Learning. In *ICLR*, pages 1928–1937.

Aaron Nicolson, Jason Dowling, and Bevan Koopman. 2023. Improving chest X-ray report generation by leveraging warm starting. *Artificial Intelligence in Medicine*, 144:102633.

Aaron Nicolson, Jason Dowling, and Bevan Koopman. 2024a. Longitudinal Data and a Semantic Similarity Reward for Chest X-Ray Report Generation. arXiv:2307.09758 [cs].

Aaron Nicolson, Shengyao Zhuang, Jason Dowling, and Bevan Koopman. 2024b. The Impact of Auxiliary Patient Data on Automated Chest X-Ray Report Generation and How to Incorporate It. ArXiv:2406.13181 [cs].

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. BLEU: a method for automatic evaluation of machine translation. In *ACL*, page 311.

Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-Critical Sequence Training for Image Captioning. In *CVPR*, pages 1179–1195.

Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Ng, and Matthew Lungren. 2020. Combining Automatic Labelers and Expert Annotations for Accurate Radiology Report Labeling Using BERT. In *EMNLP*, pages 1500–1519.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and Efficient Foundation Language Models. arXiv:2302.13971 [cs].

Maria De La Iglesia Vayá, Jose Manuel Saborit, Joaquim Angel Montell, Antonio Pertusa, Aurelia Bustos, Miguel Cazorla, Joaquin Galant, Xavier Barber, Domingo Orozco-Beltrán, Francisco García-García, et al. 2020. BIMCV COVID-19+: a large annotated dataset of RX and CT images from COVID-19 patients. *arXiv:2006.01174 [eess.IV]*.

Ronald J. Williams and David Zipser. 1989. A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. *Neural Computation*, 1(2):270–280.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the First Shared Task on Clinical Text Generation: RRG24 and "Discharge Me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. In *ICLR*.

WisPerMed at “Discharge Me!”: Advancing Text Generation in Healthcare with Large Language Models, Dynamic Expert Selection, and Priming Techniques on MIMIC-IV

Hendrik Damm[1,2], **Tabea M. G. Pakull**[3,1], **Bahadır Eryılmaz**[4,5],
Helmut Becker[4], **Ahmad Idrissi-Yaghir**[1,2], **Henning Schäfer**[3,1],
Sergej Schultenkämper[6], and **Christoph M. Friedrich**[1,2]

[1] Department of Computer Science, University of Applied Sciences and Arts Dortmund,
Dortmund, Germany

[2] Institute for Medical Informatics, Biometry and Epidemiology (IMIBE),
University Hospital Essen, Essen, Germany

[3] Institute for Transfusion Medicine, University Hospital Essen, Essen Germany

[4] Institute for AI in Medicine (IKIM), University Hospital Essen, Essen Germany

[5] Institute of Diagnostic and Interventional Radiology and Neuroradiology,
University Hospital Essen, Essen, Germany

[6] Bielefeld University of Applied Sciences and Arts, Bielefeld, Germany

{hendrik.damm, christoph.friedrich}@fh-dortmund.de

Abstract

This study aims to leverage state of the art language models to automate generating the “Brief Hospital Course” and “Discharge Instructions” sections of Discharge Summaries from the MIMIC-IV dataset, reducing clinicians’ administrative workload. We investigate how automation can improve documentation accuracy, alleviate clinician burnout, and enhance operational efficacy in healthcare facilities. This research was conducted within our participation in the Shared Task Discharge Me! at BioNLP @ ACL 2024. Various strategies were employed, including Few-Shot learning, instruction tuning, and Dynamic Expert Selection (DES), to develop models capable of generating the required text sections. Utilizing an additional clinical domain-specific dataset demonstrated substantial potential to enhance clinical language processing. The DES method, which optimizes the selection of text outputs from multiple predictions, proved to be especially effective. It achieved the highest overall score of 0.332 in the competition, surpassing single-model outputs. This finding suggests that advanced deep learning methods in combination with DES can effectively automate parts of electronic health record documentation. These advancements could enhance patient care by freeing clinician time for patient interactions. The integration of text selection strategies represents a promising avenue for further research.

1 Introduction

Clinical notes in electronic health records (EHRs) are used by clinicians to document patient progress in free-text format. These notes typically include the patient’s experiences, symptoms, findings, diagnoses, and details of procedures and interventions performed. They serve as the foundation for Discharge Summaries (DS), which contain a section with concise overviews of the entire hospital encounter known as Brief Hospital Course (BHC) (Searle et al., 2023). They are embedded in the DS and are written by senior physicians who are responsible for the patient’s overall care. In addition to BHC, DS also includes Discharge Instructions (DI), which are detailed guidelines provided to patients regarding their post-hospital care. These instructions cover the patient’s ongoing care, such as medication instructions, follow-up appointments, and any necessary lifestyle adjustments to ensure proper recovery. Discharge Instructions are designed to facilitate a smooth transition from hospital care to home care and to prevent readmissions. Writing such summaries (BHC) and instructions (DI) can be time-consuming and tedious. Consequently, physicians often spend a big portion of their clinical day dedicated to EHR documentation and desk work (Sinsky et al., 2016).

This paper presents WisPerMed’s contribution to the Shared Task Discharge Me! (Xu et al., 2024b), which is part of BioNLP @ ACL 2024. This Shared Task aims to ease the administrative burden on clinicians by developing automated methods to generate critical sections in DS, specifically the “Brief Hospital Course” and “Discharge Instruction”. Automating the creation of these sections has the potential to improve documentation accuracy, reduce

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 105–121
August 16, 2024. ©2024 Association for Computational Linguistics

clinician burnout, and ultimately optimize the processes in healthcare facilities (Patel and Lam, 2023) by allowing clinicians to allocate more time toward direct patient care.

Our work focuses on designing and implementing various innovative approaches to overcome this challenge and contribute to the overall goals of the Shared Task.

2 Dataset

The dataset (Xu, 2024) provided for this Shared Task utilizes the MIMIC-IV (Medical Information Mart for Intensive Care) database (Johnson et al., 2023a,b). MIMIC-IV is a publicly available database sourced from the EHR of the Beth Israel Deaconess Medical Center and is accessible on PhysioNet (Goldberger et al., 2000).

The task dataset is divided into four subsets: a training set consisting of 68,785 samples, a validation set containing 14,719 samples, a phase I testing set with 14,702 samples, and a phase II testing set comprising 10,962 samples. Each subset includes DS that are organized into various sections. All records contain two mandatory sections: “Brief Hospital Course” and “Discharge Instructions”. The BHC section typically provides an overview of the patient’s treatment and progress during their hospital stay and precedes the DI section. These DI summarize post-hospitalization care instructions and are positioned at the conclusion of the summary.

The challenge organizers provided a regular expression (regex) query to extract these two sections from the DS. The regex query ensures that the relevant information is accurately identified and separated from the rest of the DS content.

For the remainder of this paper, any reference to the “Discharge Summary” (DS) will exclude the target sections, BHC or DI.

3 Evaluation

The submissions to the Shared Task were evaluated using eight metrics, which assess the relevance and factuality of the generated target. These metrics include Bilingual Evaluation Understudy (BLEU-4) (Papineni et al., 2002), Recall-Oriented Understudy for Gisting Evaluation (ROUGE-1, ROUGE-2, ROUGE-L) (Lin, 2004), BERTScore (Zhang et al., 2020), Metric for Evaluation of Translation with Explicit Ordering (METEOR) (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and Medical Concept (MEDCON) (Yim et al., 2023). The overall score was calculated by averaging the scores across these eight metrics. In addition to these evaluation metrics, readability scoring metrics were also investigated and utilized in some of the developed approaches.

After the conclusion of the competition, submissions from the highest-performing teams, determined by the overall score, were evaluated by a panel of clinicians [1]. The generated sections were assessed based on their completeness, correctness, readability, and overall comparison to the reference text. These criteria were evaluated on a scale ranging from 1 to 5, where 1 signifies performance that is considerably worse than the reference text, and 5 indicates performance that is considerably better than the reference text. Three independent clinicians scored 25 DI and 25 BHC texts from each team, using the same DS.

3.1 Relevance

Relevance was evaluated using BLEU-4, ROUGE-1, ROUGE-2, ROUGE-L and BERTScore. BLEU-4 measures the precision of 4-gram matches between the generated target and reference text, providing a quantitative measure of how closely the generated target matches the reference in terms of specific sequences of words. The ROUGE metrics measure the overlap of n-grams between the target and reference texts, providing a quantifiable measure of content overlap. Furthermore, BERTScore leverages contextual embeddings to assess the semantic similarity between texts by utilizing pretrained language models such as BERT (Devlin et al., 2019). In this Shared Task, the distilBERT model (Sanh et al., 2019), a lightweight and efficient variant of BERT, was used for the BERTScore evaluation.

3.2 Factuality

Factuality in text generation was assessed using AlignScore and Summary Consistency (SummaC) (Laban et al., 2022). AlignScore measures how well the facts in a generated summary align with those in the source text. SummaC extends the AlignScore by considering both, the alignment and consistency of the generated target, ensuring it not only contains factual information but also maintains logical coherence with the source.

[1] https://stanford-aimi.github.io/discharge-me/ Accessed: 2024-05-17

Furthermore, METEOR score evaluates translation quality by aligning machine-generated target with reference translations, considering synonyms, stemming, and ordering. It balances precision and recall, and penalizes non-contiguous matches to more closely reflect human judgments than simpler metrics like BLEU-4. Lastly, the MEDCON score is a medical concept-based evaluation metric that uses the F1-score to measure the similarity between the Unified Medical Language System (UMLS) concept sets found in candidate and reference clinical notes, assessing their accuracy and consistency.

3.3 Readability

Readability was assessed using the Flesch-Kincaid Grade Level (FKGL) (Kincaid et al., 1975), Dale-Chall Readability Score (DCRS) (Chall and Dale, 1995), and Coleman-Liau Index (CLI) (Coleman and Liau, 1975). FKGL estimates the educational grade level of a text based on sentence length and syllable count per word. DCRS evaluates text complexity by identifying words not recognized by typical fourth graders. CLI calculates the grade level needed to understand the text based on character counts and sentence structure. According to CLI, higher scores indicate lower readability.

4 Methods

This section describes different approaches to the Shared Task. Licenses for the used models, frameworks, and additional datasets can be found in Appendix E.

4.1 Few-Shot learning

Few-shot learning (Wang et al., 2020) enables machine learning models to quickly adapt to new tasks using only a handful of training examples, reducing the need for extensive data collection. This method has shown improved performance on new tasks with minimal input. The Few-Shot approach utilized the WizardLM-2-8x22B (WizardLM-2) (Xu et al., 2024a) model, which was released by Microsoft and is an instruction-tuned version of the Mixtral-8x22B[2] model from Mistral AI. Refer to Appendix A for prompting examples.

4.2 Instruction Tuning

The process of instruction tuning (Peng et al., 2023) in natural language processing involves guiding a pre-trained large language model to follow specific instructions or prompts. Unlike traditional fine-tuning, which focuses on adapting the model to a specific task using a task-specific dataset, instruction tuning uses diverse instruction-based datasets to train the model to generate more accurate and relevant responses to a wide range of queries. This enables the model to better generalize across different tasks by understanding and following the instructions given.

For every experiment carried out, two models were trained: One to generate DI and one to generate BHC. Between the different experiments, hyperparameters were changed only slightly to make the experiments comparable (see Appendix C). As input format, the chat template recommended by the model publishers was used for training. Chat templates[3] are structured formats that guide the interaction between the user and the model. The input consisted of a System Message and the DS taken from the MIMIC-IV dataset. Example prompts are shown in the Appendix (see Appendix A). Most models were trained on a single NVIDIA H100 80GB using the unsloth[4] framework. Only Phi-3-Mini-128K-Instruct (Abdin et al., 2024) was trained on three NVIDIA H100 80GB. It was necessary to choose Large Language Models that are capable of handling long sequences. The average DS length is about 1,775 words or 4,243 tokens, using the Mistral-7B-Instruct-v0.2 (Jiang et al., 2023) tokenizer. All models were trained with Low-Rank Adaptation (LoRA) (Hu et al., 2022). The following models were evaluated: Llama-3-8B-Instruct (AI@Meta, 2024), Llama-3-70B-Instruct (AI@Meta, 2024), OpenBioLLM-70B (Ankit Pal, 2024), Phi-3-Mini-128K-Instruct, Mistral-7B-Instruct-v0.2. In the remainder of the paper, "I" stands for Instruct in the model naming convention. Please see Appendix C for the fine-tuning setup.

Besides the classical approach of model fine-tuning, an attempt was made to prime the models to improve their understanding of "clinical language". For this, the models were instruction-tuned with the Asclepius dataset before using the task-specific MIMIC-IV dataset. For this approach, Llama-8B-I and Mistral-7B-I-v0.2 were evaluated.

Asclepius is a dataset that was released by

[2] https://mistral.ai/news/mixtral-8x22b/ Accessed: 2024-05-14

[3] https://huggingface.co/docs/transformers/chat_templating Accessed: 2024-05-17

[4] https://github.com/unslothai/unsloth Accessed: 2024-05-14

Kweon et al. 2023. This dataset contains 158,000 rows of synthetical clinical notes and instruction-answer pairs. It was built on publicly available case reports, extracted from biomedical lectures, and then transformed into clinical notes. instruction-answer pairs were built using ChatGPT-3.5-Turbo (OpenAI, 2023).

4.3 MIMIC Section Identification

MIMIC Section Identification (MIMIC-SID) (Landes et al., 2022, 2023) is a framework used for automatically classifying sections within unstructured clinical texts, such as patient medical records. It recognizes and defines different sections of text based on their content and context. This is particularly useful in the medical domain. Documents such as DS contain distinct sections (e.g., diagnosis, treatment, patient history) that need accurate identification for effective information retrieval and processing.

Utilizing MIMIC-SID (see Figure 1), the most important sections for the target text were identified by calculating the average BERTScore (with distilBERT) between the extracted section and the target section. The text was then ordered based on relevance, from highest to lowest BERTScore, and truncated after 2,000 words. This method assumes that relevant parts are already found at the beginning of the text, and less relevant parts would be cut out. To compare this approach to a more standardized setting, the unaltered input text was also truncated to 2,000 words. This results in two training schemes: one with 2,000 words of reordered text and one with 2,000 words of the original text.

4.4 Hyperparameters

The quality of the generated targets is strongly influenced by the inference parameters employed. The Meta-Llama-3-8B-Instruct model was utilized to establish decoding strategies for the Shared Task, specifically adopting the proposed methods by (Minaee et al., 2024). Three experimental runs were conducted to examine their influence on text generation quality, each employing these decoding strategies in different configurations. The configurations and their respective parameters are detailed in Table 1.

4.5 Dynamic Expert Selection

As final approaches, five different Dynamic Expert Selections (DES) were constructed. For each DES, a set of models was pre-selected to serve as experts. Each model generates DI and BHC for a DS, and then an expert model is selected whose text is included in the submission.

Readability and factuality scores are calculated to select the expert model. The readability scores

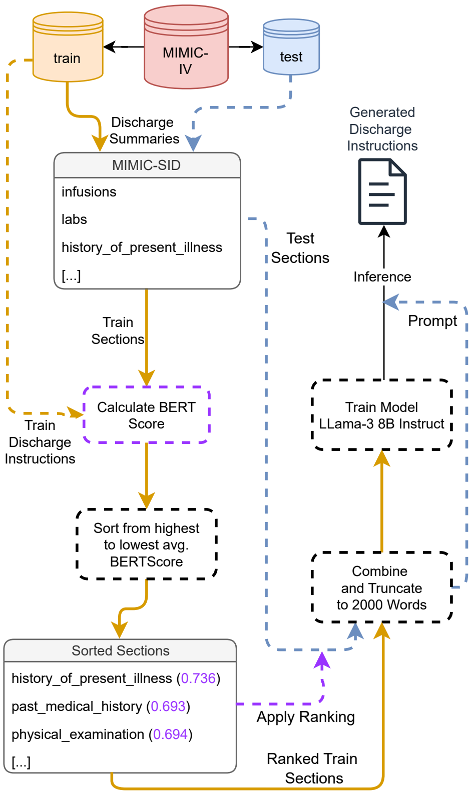

train
MIMIC-IV
test
Discharge Summaries
MIMIC-SID
infusions
labs
history_of_present_illness
[...]
Generated Discharge Instructions
Test Sections
Inference
Prompt
Train Sections
Calculate BERT Score
Train Discharge Instructions
Train Model LLama-3 8B Instruct
Sort from highest to lowest avg. BERTScore
Combine and Truncate to 2000 Words
Sorted Sections
history_of_present_illness (0.736)
past_medical_history (0.693)
physical_examination (0.694)
[...]
Apply Ranking
Ranked Train Sections

Figure 1: This workflow, exemplified by DI, is applied to BHC in the same way. With MIMIC-SID the dataset is divided into up to 50 sections. For each training section, the average BERTScore is computed using the target text as a reference. The sections are then ranked from highest to lowest BERTScore, and this ranking is applied to both the training and testing DS. The ranked training dataset is used to train the Llama-3-8B-I model. Subsequently, the ranked testing dataset is presented to the model in the form of prompts to generate DI outputs.

Parameter	Config 1	Config 2	Config 3
do_sample	False	False	True
RP	1.2	1	1
NNS	3	∞	∞
ERP	1	1.2	1
temp	0	0	0.6
top_p	0	0	0.9

Table 1: Configuration Parameters for Inference Runs. RP stands for *repetition_penalty*, NNS stands for *non_repeat_ngram_size*, and ERP stands for *encoder_repetition_penalty*.

can be calculated without any reference text, and the factuality scores use the entire DS as a reference. Because they do not use the target texts, these scores are referred to as pre-calculated scores.

Additionally, the validation set was used to compute the pre-calculated scores for the generated targets of the Mistral-7B-I-v0.2 + Asclepius model. Furthermore, the overall scores (all challenge evaluation scores) based on the target texts were determined on the validation set. Then, the correlations between the pre-calculated scores and the overall scores were examined. Figure 2 shows these correlations as a heatmap. Taking into account these correlations, DES 1-4 were constructed. For DES 5, the lengths of the generated targets were considered instead of scores as the selection criterion. This decision was based on the observation that, particularly in longer texts, models exhibit signs of hallucination or the generation of repetitive content.

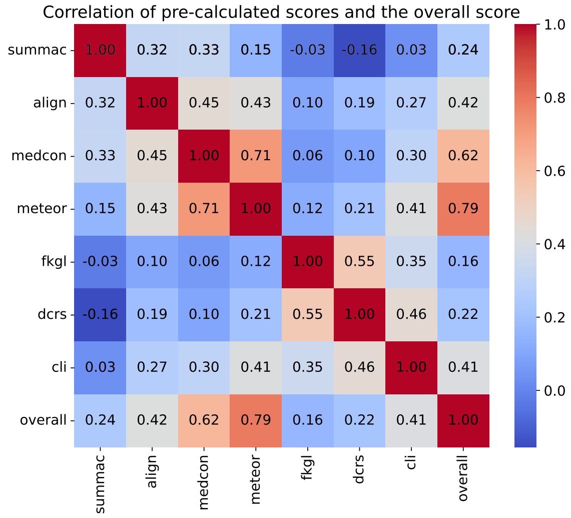
When compiling a DES, the pre-calculated scores of all included models for a DS are subjected to a min-max normalization. This means normalization over all available models. These normalized scores are then multiplied by selected weights. The model with the highest average of all normalized and weighted scores is selected as the expert for that DS.

DES 1 This DES was optimized for MEDCON and METEOR with a weight of $\frac{1}{2}$ each, as these two metrics exhibited the strongest correlation with a higher overall score. The final submission file included 6,407 texts from Mistral-7B-I-v0.2 + Asclepius, 1,210 texts from Llama-3-8B-I with greedy decoding (Minaee et al., 2024), 2,815 texts from Llama-3-8B + Asclepius, 5,600 texts from Llama-3-70B-I, 4,112 from OpenBioLLM-70B, and 1,780 from WizardLM-2.

DES 2 This DES was optimized for MEDCON and METEOR as measures for factuality, and CLI score for readability with the weights $\frac{2}{5}$ for both MEDCON and METEOR, and $\frac{1}{5}$ for CLI. The final submission file included a total of 3,471 texts from the Mistral-7B-I-v0.2 + Asclepius model, 4,374 texts from Llama-3-8B-I + Asclepius, 5,108 texts from Llama-3-70B-I, and 5,971 texts from OpenBioLLM-70B.

DES 3 This DES was optimized for all readability metrics (FKGL, DCRS, and CLI) for DI, and additionally MEDCON, METEOR, and AlignScore as factuality metrics for both text types. For the DI, all readability metrics were assigned a weight of $-\frac{1}{9}$, and the factuality metrics were assigned a weight of $\frac{2}{9}$. For the BHC, all factuality metrics were weighted with $\frac{1}{3}$. The final submission file included 5,120 texts from Mistral-7B-I-v0.2 + Asclepius, 4,211 texts from Llama-3-8B-I + Asclepius, 5,435 texts from Llama-3-70B-I, and 7,158 texts from OpenBioLLM-70B.

DES 4 This DES used only Mistral-7B-I-v0.2 models and the values of correlation between the


Correlation of pre-calculated scores and the overall score

	summac	align	medcon	meteor	fkgl	dcrs	cli	overall
summac	1.00	0.32	0.33	0.15	-0.03	-0.16	0.03	0.24
align	0.32	1.00	0.45	0.43	0.10	0.19	0.27	0.42
medcon	0.33	0.45	1.00	0.71	0.06	0.10	0.30	0.62
meteor	0.15	0.43	0.71	1.00	0.12	0.21	0.41	0.79
fkgl	-0.03	0.10	0.06	0.12	1.00	0.55	0.35	0.16
dcrs	-0.16	0.19	0.10	0.21	0.55	1.00	0.46	0.22
cli	0.03	0.27	0.30	0.41	0.35	0.46	1.00	0.41
overall	0.24	0.42	0.62	0.79	0.16	0.22	0.41	1.00

Figure 2: Heatmap of the Pearson correlations between pre-calculated scores and the overall score on the validation dataset. The pre-calculated scores include factuality scores (SummaC, AlignScore, MEDCON and METEOR), which are calculated for the generated targets of the Mistralv2 + Asclepius model with the whole DS as the reference, and readability scores (FKGL, DCRS and CLI).

Example given:

[...Non-repetitive text]
-Please check LFTs weekly for the next month.
-Please continue to monitor for signs of refeeding hyperglycemiAsclepius
-Please continue to monitor for signs of refeeding hypocalcemiAsclepius
[Continues Repetition...]

Figure 3: Example of repetitive and hallucinated DI output generated by Llama-3-8B-I. The words hyperglycemia and hypocalcemia are very similar but only one of them should be in the generated targets. The other one was not mentioned in the DS.

scores calculated on the generated targets against the whole DS and the overall score on the validation dataset as weights. The final submission file included 7,912 texts from Mistral-7B-I-v0.2 + Asclepius, 6,485 texts from the Mistral-7B-I-v0.2 + Asclepius model, which was further fine-tuned on the validation dataset, and 7,527 texts from Mistral-7B-I-v0.2.

DES 5 This DES considers lengths of texts instead of weighted metrics. DI in the training dataset has an average word count of approximately 196.3, whereas BHCs have an average word count of approximately 327.6. Models trained on these texts, therefore, tend to generate shorter texts for DI and longer texts for BHC. To mitigate the impact of hallucinations at the end of lengthy texts (e.g Figure 3), a strategy of preferably selecting shorter texts with the DES was adopted. The objective of the strategy was to initially rank the models based on their overall scores. Subsequently, for each DS, the text from the first model that has a word count within the range of 100 to 180 words is selected. If no model had generated a text with a word count within this range, the text with the minimum word count was selected. However, the text could not be shorter than 70 words. In the case that no text met these criteria, the text from the highest-ranked model remained.

5 Results

This section describes results of the evaluation of the developed models using the metrics described in section 3. The evaluation done by clinicians can be seen in Table 3.

5.1 Automatic Evaluation

The final scores for the Shared Task, provided by the organizers, are shown in Table 2. The table presents the approaches from the most general to the most specific, beginning with the baseline model, followed by the Few-Shot model, the instruction-tuned models, the instruction-tuned models primed with Asclepius, the MIMIC section-based approaches, and the various DES variants. The top competitors by overall score are highlighted for comparison. All inference runs for the final results were conducted with an optimized decoding strategy (temp=0.6, top_p=0.9) as described in section 4.4. Based on its high evaluation scores, Configuration 3 was chosen as the standard parameter setting, which can be seen in Table 1.

WizardLM-2, despite not being fine-tuned on the training data, surpassed the baseline with an overall score of 0.195.

Among the fine-tuned models, Llama-3-70B-I led with a score of 0.300, followed by Mistral-7B-I-v0.2 at 0.289, which has way less parameters yet outperformed several larger models, including Llama3-8B-I. Even though the OpenBioLLM-70B has been adapted for clinical use, it underperformed when compared to other models. The Phi-3-mini-128k-I (3.8 billion parameters) model matched the performance of larger models, such as the Llama-8B-I, demonstrating the efficiency of models with less parameters.

Among the configurations, incorporating the Asclepius dataset into Mistral-7B-I-v0.2 made it clearly outperform Llama3-8B-I. Excluding the DES approaches, this combination achieved the highest performance of all models.

The MIMIC-SID approaches with a shorter context length of 2,000 words displayed weak performances.

Different parameters on the Llama-8B-I, including greedy decoding and an ERP (encoder_repetition_penalty), yielded lower scores compared to setups utilizing sampling and temperature adjustments.

In the Dynamic Expert Selection category, DES 1 focused on MEDCON and METEOR scores but did not surpass the individual fine-tuned models. DES 2 achieved the highest BERTScore and AlignScore, which represent relevance and factuality, respectively. Reaching an overall score of 0.311, this DES outperformed all individual fine-tuned models. DES 3, aimed at lowering readability metrics, scored 0.296, performing better than DES 1 but lagging behind others. DES 4, using correlation values for optimization, showed negligible improvements. DES 5 achieved the highest overall score of 0.332 and topped the leaderboard by limiting text length. The approach achieved the highest scores in all metrics, except for a slightly lower BERTScore and AlignScore.

5.2 Clinical Evaluation

The ranking order of the first six teams did not change when comparing the automatic with the clinicians' evaluation results (see Table 2 and Table 3). The evaluated BHC were ranked the best over all aspects. The readability and holistic evaluation of the BHC were notably superior to that of the other teams. However, the DI scores were compa-

Model	Ovr.	BLEU	R-1	R-2	R-L	BERT	MET	Align	MED
Baseline									
Challenge Baseline	0.102	0.015	0.126	0.051	0.113	0.138	0.098	0.167	0.121
Few-Shot learning									
WizardLM-2 8x22B	0.195	0.017	0.257	0.074	0.158	0.331	0.310	0.193	0.218
Instruction-tuned									
Llama-3-8B-I	0.253	0.053	0.331	0.107	0.241	0.392	0.235	0.320	0.348
Mistral-7B-I-v0.2	0.289	0.101	0.371	0.122	0.252	0.416	**0.375**	0.293	0.380
Llama-3-70B-I	**0.300**	**0.112**	0.367	**0.141**	**0.260**	**0.437**	0.347	0.334	**0.401**
OpenBioLLM-70B	0.285	0.084	0.376	0.127	0.248	0.421	0.307	**0.337**	0.383
Phi-3-mini-128k-I	0.254	0.062	**0.347**	0.128	0.217	0.359	0.310	0.275	0.330
Instruction-tuned + Asclepius (A.)									
Llama-3-8B-I + A.	0.302	0.107	0.388	**0.150**	**0.275**	0.432	0.350	0.311	0.403
Mistral-7B-I-v0.2 + A.	**0.307**	**0.120**	**0.390**	0.140	0.258	**0.434**	**0.391**	**0.320**	**0.404**
MIMIC Section Identification									
Llama-3-8B-I 2k	0.209	0.022	0.263	0.054	0.171	0.326	**0.199**	**0.355**	0.280
Llama-3-8B-I R 2k	**0.216**	**0.026**	**0.292**	**0.073**	**0.191**	**0.351**	0.186	0.306	**0.304**
Hyperparameter									
Llama-3-8B-I Greedy	0.192	0.018	0.274	0.043	0.147	0.314	**0.221**	0.281	0.241
Llama-3-8B-I ERP	**0.238**	**0.032**	**0.348**	**0.093**	**0.228**	**0.372**	**0.221**	**0.307**	**0.300**
Dynamic Expert Selection									
DES 1	0.277	0.097	0.329	0.121	0.217	0.417	0.339	0.319	0.374
DES 2	0.311	0.110	0.414	0.151	0.273	**0.439**	0.351	**0.344**	0.406
DES 3	0.296	0.108	0.366	0.128	0.242	0.435	0.352	0.335	0.400
DES 4	0.297	0.112	0.371	0.127	0.244	0.426	0.379	0.320	0.396
DES 5	**0.332**	**0.124**	**0.453**	**0.201**	**0.308**	0.438	**0.403**	0.315	**0.411**
Top 5 Competitors									
HarmonAI Lab Yale	**0.300**	**0.106**	0.423	0.180	**0.284**	**0.412**	0.381	0.265	0.353
aehrc	0.297	0.097	0.414	**0.192**	**0.284**	0.383	**0.398**	0.274	0.332
EPFL-MAKE	0.289	0.098	**0.444**	0.155	0.262	0.399	0.336	0.255	**0.360**
UF-HOBI	0.286	0.102	0.401	0.174	0.275	0.395	0.289	**0.296**	0.355
de ehren	0.284	0.097	0.404	0.166	0.265	0.389	0.376	0.231	0.339

Table 2: Summary of model performance across different experimental settings. Each section represents a distinct approach: Baseline, Few-Shot Learning, instruction-tuned, instruction-tuned + Asclepius, MIMIC-SID (2k for truncation to 2k words and R for reordering the subsections in the text from most to least relevant according to BERTScore), Hyperparameter, and DES, showcasing respective strategies to address the challenge. *I* indicates that the instruction version of the model was used. Furthermore, the Top 5 runs from other challenge participants are included. Metrics include overall score (Ovr.), BLEU-4 (BLEU), ROUGE-1 (R-1), ROUGE-2 (R-2), and ROUGE-L (R-L), BERTScore (BERT), METEOR (MET), AlignScore (Align) and MEDCON (MED). Bold scores indicate the best performance in each category, with underlined bold scores highlighting the top overall scores across all experiments.

rable to those of other teams and did not achieve the highest score in any of the evaluated aspects.

In order to compare automated evaluation results with clinicians' assessments, it was necessary to normalize the scores on a scale from 0 to 1. Note that readability was not compared, as clinicians did not rate the readability of DI texts, nor did the challenge metrics include readability scores. Figure 4 illustrates that clinicians tend to assign higher scores than the automated metrics in their evaluation approach. However, the holistic evaluation aligns more closely with the overall automated scores.

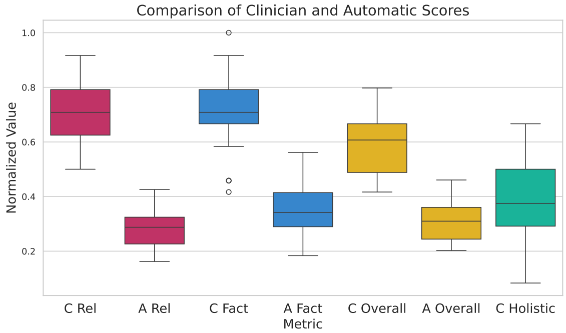


Figure 4: Boxplot of Average Clinician Scores and Average Metric Scores. C stands for Clinician and A stands for the scores caluclate with the challenge metrics. Here A Relevance includes ROUGE-1, ROUGE-2, ROUGE-L, BertScore and BLEU-4. A Factuality includes AlignScore METEOR and MEDCON.

6 Discussion

Despite being one of the less robust models evaluated, WizardLM-2 exceeded the established baseline, showing its effectiveness in a Few-Shot learning context. With minimal training examples, the model still produced high-quality texts, according to the metrics, highlighting the potential of Few-Shot learning in enhancing performance metrics.

The performance of instruction-tuned models revealed mixed outcomes. Despite being a specialized adaptation of the Llama-3-70B-I model tailored for medical contexts, OpenBioLLM-70B underperformed in relation to its base model. This behavior was unexpected, considering its design to enhance relevance and accuracy in clinical applications. Conversely, the Mistral-7B-I-v0.2 model demonstrated impressive capabilities, outperforming both the larger OpenBioLLM-70B and the Llama-8B-I models. This highlights the effectiveness of Mistral-7B-I-v0.2 in handling complex medical text generation and summarization tasks despite its smaller size. In contrast to its reputation as one of the most promising state-of-the-art open-source LLMs, the Llama-3 models have been found to be less effective in this challenge. This is on par with the findings from LMSYS chatbot arena (Chiang et al., 2024) where LLama-3 models showed the weakest performance compared to other state-of-the-art models on the task of summarization (Dunlap et al., 2024).

Using the Asclepius dataset for priming substantially improved model performance during the fine-tuning phases. For instance, the Llama-8B-I model's score rose from 0.253 to 0.302, and the Mistral model's performance increased from 0.289 to 0.307. Notably, the Mistral-7B-I-v0.2 + Asclepius model was the top performer in the challenge, aside from DES approaches. This underscores the benefits of further training models with specialized datasets to enhance accuracy and relevance in domain-specific tasks.

The reordering of sections within the MIMIC-SID approach moderately enhanced overall model performance, demonstrating that prioritizing the most relevant sections can be beneficial. However, it is important to note that metrics sensitive to text order, such as METEOR and AlignScore, experienced a decline. This suggests that while reordering can improve general outcomes by emphasizing key information, it may simultaneously compromise the sequential integrity of the text. Therefore, this strategy confirms the utility of structurally optimizing input for task-specific relevance, albeit with some trade-offs in textual coherence.

Exploration of hyperparameter settings revealed that more complex configurations did not yield superior results. The basic approach, utilizing *do_sample=True*, *temp=0.6*, and *top_p=0.9*, consistently outperformed other tested configurations, including those with greedy decoding and encoder repetition penalties. This emphasizes the efficacy of maintaining simpler hyperparameter settings for stable and high-quality text generation. Additional complexity in parameter tuning did not always correlate with improved model performance.

The DES that relied on the pre-calculated scores had varying effects on the metrics evaluated. Using MEDCON and METEOR in combination with CLI improved the results, whereas choosing the correlation as weights resulted in no improvement. A possible reason might be that the pre-calculated scores were only calculated on the entire DS and

		BHC				DI		
Team	Avg.	Comp	Corr	Read	Hol.	Comp	Corr	Hol.
WisPerMed	**3.375**	**3.667**	**3.667**	**3.373**	**2.440**	3.947	4.000	2.533
HarmonAI Lab at Yale	2.903	3.520	2.587	2.107	1.533	**4.267**	3.947	2.360
aehrc	2.785	2.307	3.053	1.960	1.093	3.907	**4.547**	**2.627**
EPFL-MAKE	2.720	3.293	2.827	2.533	1.653	3.453	3.413	1.867
UF-HOBI	2.579	2.480	3.360	2.707	1.413	3.013	3.293	1.787
de ehren	2.335	2.280	2.987	2.680	1.120	2.813	3.053	1.413

Table 3: BHC and DI Metrics for Teams by clinicans. In this Table Avg. stands for Average, Comp stands for Comperability, Corr stands for Correctness, Read stands for Readability, and Hol stands for Holistic.

not on the target text, as in the final evaluation. It may also be that the correlations are not always sufficient, and a more elaborate association analysis is needed.

Consequently, the best overall score of all DES was achieved by the approach limiting the text length, suggesting that hallucinations and repetitive sequences have a measurable impact on text quality.

The manual evaluation seen in Table 3 indicates that the holistic approach by clinicians is comparable to the automated metrics, thus reconfirming the effectiveness of the metrics used in the competition. The lower scores for the DI may be caused by information loss or distortion due to the simplification.

7 Conclusion

The research identified several opportunities for future investigation that may enhance the performance and utility of the discussed models. Initially, due to the extensive size of the training dataset and the constraints imposed by the context length of input texts, each model was trained for a maximum of only three epochs. Therefore, extending the training duration may provide improvements and merits further exploration.

Moreover, alterations to inference parameters have demonstrated notable effects on model outputs. For example, employing the ERP parameter, while maintaining other settings constant resulted in a degradation of performance metrics (from 0.253 to 0.238 overall score). This suggests a systematic evaluation of inference parameters could further enhance model output.

Additionally, priming the model has substantially improved results. Investigating additional datasets for priming purposes could further optimize model performance and expand its applicability across diverse textual tasks. This could be a promising direction for future research efforts. Further opportunities lie in optimizing section reordering to balance task-specific relevance while maintaining text coherence.

The winning approach, evaluated by the automatic and clinicans' evaluation, a DES, achieved the highest overall score. This suggests that generating multiple outputs and developing methodologies to select the optimal text may further improve performance. Therefore, exploring various DES techniques and selection criteria is a field for further research.

The clinicians' evaluation added valuable insights, and the automatic scores demonstrated high robustness and strong alignment with the manual assessments. This alignment indicates that the manual evaluation, even on a small subset, effectively validates the reliability of the generated texts.

Lastly, efforts to enhance the quality of medical machine learning algorithms are ongoing, along with a responsibility to report the environmental impact of the research. In this study, the total energy consumption for training and inference is estimated with 1,552.10 kWh, resulting in 591.35 kg CO_2 emission. Detailed information is provided in Appendix D.

Limitations

For the model training, only the entire discharge summaries were utilized, while the provided radiology reports and ICD 9/10 codes were not included. The decision to exclude these additional documents might have limited the comprehensiveness of our models. Future research should consider incorporating these documents to potentially improve model accuracy and contextual understanding.

Moreover, each model was trained for a maximum of three epochs due to the context length constraints of input texts. Extending the training duration could potentially enhance performance and merit further exploration.

Additionally, the influence of inference parameters on model outputs is notable. Systematic evaluation of these parameters is needed, as variations can measurably affect performance metrics.

Furthermore, the study did not employ advanced preprocessing or postprocessing techniques, which could substantially enhance the reliability and accuracy of the generated texts by mitigating issues such as non-factual content generation ("hallucinations"). Notably, DES 5 considered text length, which may indirectly reduce hallucinations. However, this approach does not explicitly address the issue and therefore cannot ensure their complete avoidance.

The Asclepius dataset, being synthetic and partly based on the MIMIC-III dataset, may introduce data redundancy or leakage, potentially impacting model robustness and generalizability. Future work should explore advanced data validation techniques or alternative dataset creation methodologies to mitigate these issues.

Lastly, while priming models with specialized datasets showed substantial improvements, further investigation into additional datasets for priming could optimize model performance and expand applicability across diverse textual tasks.

Acknowledgement

The work of Hendrik Damm, Tabea Pakull, Bahadır Eryılmaz, Helmut Becker, Ahmad Idrissi-Yaghir and Henning Schäfer was funded by a PhD grant from the DFG Research Training Group 2535 *Knowledge-and data-based personalization of medicine at the point of care (WisPerMed)*.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Caio César Teodoro Mendes, Weizhu Chen, Vishrav Chaudhary, Parul Chopra, Allie Del Giorno, Gustavo de Rosa, Matthew Dixon, Ronen Eldan, Dan Iter, Amit Garg, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Jamie Huynh, Mojan Javaheripi, Xin Jin, Piero Kauffmann, Nikos Karampatziakis, Dongwoo Kim, Mahoud Khademi, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Chen Liang, Weishung Liu, Eric Lin, Zeqi Lin, Piyush Madan, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Xia Song, Masahiro Tanaka, Xin Wang, Rachel Ward, Guanhua Wang, Philipp Witte, Michael Wyatt, Can Xu, Jiahang Xu, Sonali Yadav, Fan Yang, Ziyi Yang, Donghan Yu, Chengruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

AI@Meta. 2024. Llama 3. https://github.com/meta-llama/llama3/. Accessed: 2024-05-15.

Malaikannan Sankarasubbu Ankit Pal. 2024. OpenBioLLMs: Advancing open-source large language models for healthcare and life sciences. https://huggingface.co/aaditya/OpenBioLLM-Llama3-70B. Accessed: 2024-05-15.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

J.S. Chall and E. Dale. 1995. *Readability Revisited: The New Dale-Chall Readability Formula*, volume 1. Brookline Books.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. Chatbot arena: An open platform for evaluating llms by human preference. *Preprint*, arXiv:2403.04132.

Meri Coleman and Ta Lin Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283–284.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Lisa Dunlap, Evan Frick, Tianle Li, Isaac Ong, Joseph E. Gonzalez, and Wei-Lin Chiang. 2024. What's up with llama 3? arena data analysis. Accessed: 2024-05-15.

Ary L. Goldberger, Luis A. N. Amaral, Leon Glass, Jeffrey M. Hausdorff, Plamen Ch. Ivanov, Roger G.

Mark, Joseph E. Mietus, George B. Moody, Chung-Kang Peng, and H. Eugene Stanley. 2000. Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B. *Preprint*, arXiv:2310.06825.

A. Johnson, L. Bulgarelli, T. Pollard, S. Horng, L. A. Celi, and R. Mark. 2023a. MIMIC-IV.

Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, Liwei H. Lehman, Leo A. Celi, and Roger G. Mark. 2023b. Mimic-iv, a freely accessible electronic health record dataset. *Scientific Data*, 10:1.

J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Sunjun Kweon, Junu Kim, Jiyoun Kim, Sujeong Im, Eunbyeol Cho, Seongsu Bae, Jungwoo Oh, Gyubok Lee, Jong Hak Moon, Seng Chan You, Seungjin Baek, Chang Hoon Han, Yoon Bin Jung, Yohan Jo, and Edward Choi. 2023. Publicly shareable clinical large language model built on synthetic clinical notes. *Preprint*, arXiv:2309.00237.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting nli-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. 2019. Quantifying the carbon emissions of machine learning. *CoRR*, abs/1910.09700.

Paul Landes, Barbara Di Eugenio, and Cornelia Caragea. 2023. DeepZensols: A deep learning natural language processing framework for experimentation and reproducibility. In *Proceedings of the 3rd Workshop for Natural Language Processing Open Source Software (NLP-OSS 2023)*, pages 141–146, Singapore, Singapore. Empirical Methods in Natural Language Processing.

Paul Landes, Kunal Patel, Sean S. Huang, Adam Webb, Barbara Di Eugenio, and Cornelia Caragea. 2022. A new public corpus for clinical section identification: MedSecId. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3709–3721, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large language models: A survey. *arXiv preprint arXiv:2402.06196*.

OpenAI. 2023. ChatGPT. `https://www.openai.com`. Accessed: 2024-05-15.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

SB Patel and K Lam. 2023. Chatgpt: the future of discharge summaries? *Lancet Digit Health*, 5(3):e107–e108. Epub 2023 Feb 6.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *Preprint*, arXiv:2304.03277.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Thomas Searle, Zina Ibrahim, James Teo, and Richard J.B. Dobson. 2023. Discharge summary hospital course summarisation of in patient electronic health record text with clinical concept guided deep pre-trained transformer models. *Journal of Biomedical Informatics*, 141:104358.

Christine Sinsky, Lacey Colligan, Ling Li, Mirela Prgomet, Sam Reynolds, Lindsey Goeders, Johanna Westbrook, Michael Tutty, and George Blike. 2016. Allocation of physician time in ambulatory practice: A time and motion study in 4 specialties. *Annals of Internal Medicine*, 165(11):753.

Dennis Ulmer, Elisa Bassignana, Max Müller-Eberstein, Daniel Varab, Mike Zhang, Rob van der Goot, Christian Hardmeier, and Barbara Plank. 2022. Experimental standards for deep learning in natural language processing research. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2673–2692, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys*, 53(3).

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. 2024a. WizardLM: Empowering large pre-trained language models to follow complex instructions. In *The Twelfth International Conference on Learning Representations*.

J. Xu. 2024. Discharge me: Bionlp acl'24 shared task on streamlining discharge documentation.

Justin Xu, Andrew Johnston, Zhihong Chen, Louis Blankemeier, Maya Varma, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024b. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

W. Yim, Y. Fu, A. Ben Abacha, N. Snider, T. Lin, and M. Yetisgen. 2023. Aci-bench: A novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10(1):586.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Few-Shot Learning Prompts

This section showcases how the WizardLM-2-Model was instructed. A variety of prompts were tested, and the displayed ones (see Figure 5 and Figure 6) yielded the best results, as measured by human evaluation. Detailed instructions were provided for the DI text generation, whereas the details for the BHC generation were excluded. The BHC texts are considerably longer on average and do not follow the same pattern most of the time.

Discharge Instructions Prompt

USER: Generate a detailed discharge instruction based on the provided summary, adhering to the style of the provided examples. The instruction should comprehensively cover all aspects of the patient's care, with a total length of about 300-500 words.

Please follow the format used in previous discharge instructions:

1. Start with a polite greeting and an expression of gratitude or pleasure for having taken care of the patient.
2. Describe the reason for hospitalization succinctly.
3. Detail what occurred during the stay, including any treatments administered, patient responses, and significant changes to the patient's condition.
4. Outline clear follow-up care instructions, including medications, dietary recommendations, activity level, and scheduled follow-up visits.
5. Close with a kind farewell and additional well-wishes or reminders.

Discharge Instruction Format Example:
Dear [Patient Name],
It was a pleasure taking care of you during your hospitalization at [Hospital Name].

Why were you hospitalized?
- [Brief reason for hospitalization]

What happened while you were in the hospital?
- [Key details about treatment and patient response]
- [Any significant tests and their results]
- [Any changes to patient condition]

What should you do after you leave the hospital?
- [Medications and dosage]
- [Dietary instructions]
- [Activity recommendations]
- [Follow-up appointments]

We wish you the best in your recovery!

Sincerely,
Your [Hospital Team Name] Team

Discharge Instruction Example 1 start:
[Discharge Instruction Example from Validation set]
Discharge Instruction Example 1 end.
[· · ·]
Discharge Instruction Example 10 start:
[Discharge Instruction Example from Validation set]
Discharge Instruction Example 10 end.
Discharge Summary:
[Discharge Summary without target section]
Start with the Discharge Instructions for the Discharge Summary.
ASSISTANT:

Figure 5: Discharge Instruction Prompt for Few-Shot learning with WizradLM-2.

Brief Hospital Course Prompt

USER: Here are some Example Brief Hospital Courses.

Brief Hospital Course Example 1 start:
[Brief Hospital Course Example from Validation set]
Brief Hospital Course Example 1 end.
[· · ·]
Brief Hospital Course Example 7 start:
[Brief Hospital Course Example from Validation set]
Brief Hospital Course Example 7 end.

Now create a Brief Hospital Course in the same style as in the Examples with the information from the following Discharge Summary:
[Discharge Summary without target section]

ASSISTANT:

Figure 6: Brief Hospital Course Prompt for Few-Shot learning with WizardLM-2.

B Instruction Tuning Prompts

Figure 7 and Figure 8 show the prompts used for instruction tuning DI and BHC. The only difference between the instruction tuning and inference prompts is that the [Target Discharge Instructions] or [Target Brief Hospital Course] was left empty for inference. For each model, the recommended chat template provided by the model inventors was followed and applied. This is especially important when using the instruction version of those models.

Discharge Instructions Prompt

<SYSTEM>You are in the world's best hospital as the best doctor. You're given a patient's details summarized by your medical staff in 'Summary'. You now need to figure out the 'Discharge Instructions' for the patient. Think carefully without error, since you might endanger a patient's life, which we do not want to happen.

<User>Summary: [Discharge Summary without target section]

Discharge Instructions:

<ASSISTANT>[Target Discharge Instructions]

Figure 7: Instruction Tuning and Inference Prompt for Discharge Instructions.

Brief Hospital Course Prompt

<SYSTEM>You are in the world's best hospital as the best doctor. You're given a patient's details summarized by your medical staff in 'Summary'. You now need to figure out a 'Brief Hospital Course' for the patient. Think carefully without error, since you might endanger a patient's life, which we do not want to happen.

<USER>Summary: [Discharge Summary without target section]

Brief Hospital Course:
<ASSISTANT>[Target Brief Hospital Course]

Figure 8: Instruction Tuning and Inference Prompt for Brief Hospital Course.

C Parameter Setup

Whenever possible, hyperparameters were only changed slightly to ensure high comparability between results. The LoRA setup is detailed in Table 4. The following modules were targeted with LoRA: “q_proj”, “k_proj”, “v_proj”, “o_proj”, “gate_proj”, “up_proj”, and “down_proj”. While it is suggested[5] to use a LoRA Rank = LoRA Alpha * 2, this approach was not chosen due to VRAM efficiency considerations. For the detailed training setup, please see Table 5. All models were trained on 80GB H100s and 48GB RTX6000s. The Unsloth open-source training framework was used because it reduced VRAM usage by at least 50% and subsequently made fine-tuning runs twice as fast. This efficiency allowed training almost all models on a single GPU. The Maximum Sequence Length for the 70B models was reduced to decrease memory consumption. For Phi-3, the optimizer was changed from adamw_8bit (Loshchilov and Hutter, 2019) to paged_adamw_8bit[6] to further optimize memory usage.

Model	LR	LA	loadIn4Bit	LD	GC	DT
Llama-3-8B-I + A. Prime	16	16	true	0	true	bfloat16
Mistral-7B-I-v0.2 + A. Prime	16	16	true	0	true	bfloat16
Llama-3-8B-I	16	16	true	0	true	bfloat16
Mistral-7B-I-v0.2	16	16	true	0	true	bfloat16
Llama-3-70B-I	16	16	true	0	true	bfloat16
OpenBioLLM-70B	16	16	true	0	true	bfloat16
Phi-3-mini-128k-I	16	16	true	0	true	bfloat16
Llama-3-8B-I 2k + A	16	16	true	0	true	bfloat16
Mistral-7B-I-v0.2 + A.	16	16	true	0	true	bfloat16
Llama-3-8B-I 2k	16	16	true	0	true	bfloat16
Llama-3-8B-I R 2k	16	16	true	0	true	bfloat16

Table 4: LoRA Setup for fine-tuning. LR means LoRA Rank, LA means LoRA Alpha, LD means LoRA Dropout, GC means Gradient Checkpointing, DT means dtype. By A. Asclepius is meant. Prime means the instruction tuning runs with Asclepius.

Model	MSL	E	GAS	WS	LR	BS	O	S	WD
Llama-3-8B-I + A. P.	15,000	1	4	5	2e-4	4	adamw_8bit	linear	0.01
Mistral-7B-I-v0.2 + A. P.	15,000	1	4	5	2e-4	4	adamw_8bit	linear	0.01
Llama-3-8B-I	10,000	3	4	5	2e-4	4	adamw_8bit	linear	0.01
Mistral-7B-I-v0.2	10,000	3	4	5	2e-4	4	adamw_8bit	linear	0.01
Llama-3-70B-I	10,000	2	4	5	2e-4	2	adamw_8bit	linear	0.01
OpenBioLLM-70B	10,000	2	4	5	2e-4	2	adamw_8bit	linear	0.01
Phi-3-mini-128k-I	12,000	2	4	10	2e-4	4	p_adamw_8bit	linear	0.01
Llama-3-8B-I + A.	13,000	2	4	5	2e-4	4	adamw_8bit	linear	0.01
Mistral-7B-I-v0.2 + A.	13,000	2	4	5	2e-4	4	adamw_8bit	linear	0.01
Llama-3-8B-I 2k	6,000	3	4	5	2e-4	4	adamw_8bit	linear	0.01
Llama-3-8B-I R 2k	6,000	3	4	5	2e-4	4	adamw_8bit	linear	0.01

Table 5: MSL means Maximum Sequence Length (Tokens), E means Epochs, GAS means Gradient Accumulation Steps, WS means Warmup Steps, LR means Learning Rate, BS means Batch Size, O means Optimizer, S means Scheduler, WD means Weight Decay. By A. Asclepius is meant. Prime (P.) means the instruction tuning runs with Asclepius.

[5] https://magazine.sebastianraschka.com/p/practical-tips-for-finetuning-llms Accessed: 2024-05-17

[6] https://huggingface.co/docs/bitsandbytes/en/optimizers#paged-optimizers Accessed: 2024-05-15

D Environmental Impact

In scientific research, it is crucial to consider not only the direct results of experiments but also the broader implications and consequences of the research process. While the following environmental assessment is not directly tied to the primary results, reporting on the environmental footprint of the work is essential given the increasing global emphasis on sustainability and the environmental impact of computational practices. This perspective aligns with the findings of (Ulmer et al., 2022), emphasizing the importance of understanding and reporting the environmental consequences of experimental work.

The experiments were conducted using HPC resources located in Essen and Dortmund, Germany. The region's electricity generation has a carbon efficiency of 0.381 kgCO_2 eq/kWh[7], with approximately 41,1% [8] of the electricity being sourced from fossil fuels. To estimate the carbon footprint of our experiments, the Machine Learning Impact calculator, as presented by (Lacoste et al., 2019), is utilized. This calculator provides a comprehensive framework to quantify the carbon emissions associated with machine learning experiments, considering both the energy consumption of computational resources and the carbon efficiency of the electricity source.

Final Models	**Runtime (hours)**	**Power (Avg. Watts)**	**Energy (kWh)**	**CO_2 (kg)**
Mistral-7B-I-v0.2 BHC + A.	28.5	651.45	18.58	7.08
Mistral-7B-I-v0.2 + A. Prime	5	637	3.21	1.22
Mistral-7B-I-v0.2 DI + A.	27.4	681	18.71	7.13
Experiment runs	1,920	783.83	1,511.59	575.91
Overall	1,980.9	783.532	1,552.10	591.35

Table 6: Runtime, Energy Consumption and CO_2 Emissions for the Final models, Other Experiment Runs and Overall for All Experiments. By A. Asclepius is meant. Prime means the instruction tuning runs with Asclepius.

The carbon footprint and electricity consumption values for our optimal models, as well as for all experimental runs conducted throughout the research process presented in Table 6. The values indicate that substantial resources are expended on debugging and testing during development.

[7] https://ourworldindatAsclepiusorg/grapher/carbon-intensity-electricity?country=~DEU Accessed: 2024-05-14

[8] https://www.destatis.de/EN/Themes/Economic-Sectors-Enterprises/Energy/Production/Tables/gross-electricity-production.html Accessed: 2024-05-14

E Licenses

In Table 7 the Licenses as given by the owners of the Dataset/Framework/Model are displayed.

Dataset/Framework/Model	License
Asclepius dataset[9]	Creative Commons Attribution Non Commercial Share Alike 4.0
MIMIC-IV-Note[10]	PhysioNet Credentialed Health Data License 1.5.0
MIMIC-IV-ED[11]	PhysioNet Credentialed Health Data License 1.5.0
MIMIC-SID[12]	MIT License
unsloth[13]	Apache License Version 2.0
Mistral-7B-I-v0.2[14]	Apache License Version 2.0
Llama-3-8B-I[15]	Llama 3 Community License Agreement
Llama-3-70B-I[16]	Llama 3 Community License Agreement
OpenBioLLM-70B[17]	Llama 3 Community License Agreement
WizardLM-2 8x22B[18]	MIT License
Phi-3-mini-128k-I[19]	Apache License Version 2.0

Table 7: Licenses of the dataset, Framework and Models used for this Shared Task.

[9] https://huggingface.co/datasets/starmpcc/Asclepius-Synthetic-Clinical-Notes Accessed: 2024-05-17

[10] https://physionet.org/content/mimic-iv-note/2.2/ Accessed: 2024-05-17

[11] https://physionet.org/content/mimic-iv-ed/2.2/ Accessed: 2024-05-17

[12] https://github.com/plandes/mimicsid Accessed: 2024-05-17

[13] https://github.com/unslothai/unsloth Accessed: 2024-05-17

[14] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2 Accessed: 2024-05-17

[15] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct Accessed: 2024-05-17

[16] https://huggingface.co/meta-llama/Meta-Llama-3-70B-Instruct Accessed: 2024-05-17

[17] https://huggingface.co/aaditya/Llama3-OpenBioLLM-70B Accessed: 2024-05-17

[18] https://huggingface.co/alpindale/WizardLM-2-8x22B Accessed: 2024-05-17

[19] https://huggingface.co/microsoft/Phi-3-mini-128k-instruct Accessed: 2024-05-17

Overview of the BioLaySumm 2024 Shared Task on the Lay Summarization of Biomedical Research Articles

Tomas Goldsack[1], **Carolina Scarton**[1], **Matthew Shardlow**[3], **Chenghua Lin**[1,2]
[1]University of Sheffield, [2]University of Manchester, [3]Manchester Metropolitan University
{tgoldsack1, c.lin, c.scarton}@sheffield.ac.uk
m.shardlow@mmu.ac.uk

Abstract

This paper presents the setup and results of the second edition of the BioLaySumm shared task on the Lay Summarisation of Biomedical Research Articles, hosted at the BioNLP Workshop at ACL 2024. In this task edition, we aim to build on the first edition's success by further increasing research interest in this important task and encouraging participants to explore novel approaches that will help advance the state-of-the-art. Encouragingly, we found research interest in the task to be high, with this edition of the task attracting a total of 53 participating teams, a significant increase in engagement from the previous edition. Overall, our results show that a broad range of innovative approaches were adopted by task participants, with a predictable shift towards the use of Large Language Models (LLMs).

1 Introduction

Lay Summarisation describes the task of generating a summary of a technical or specialist text that is suitable for a non-expert audience. To achieve this goal, a good lay summary will typically focus on explaining the relevant background information alongside the significance and findings of an article, while avoiding extensive use of jargon or technical language. As such, lay summaries offer significant benefits in broadening access to technical articles that are of interest to a broad range of audiences.

Biomedical research publications, containing the latest research on prominent health-related topics, represent a perfect example of such texts. Not only are the contents of these articles relevant to other researchers working in the same domain, but often they can also be of interest to researchers in related domains, medical practitioners, and even members of the public (e.g., those affected by an illness/disease being studied). In this scenario, the lay summary is an essential tool in allowing these secondary audiences, who don't possess the expertise required to interpret the full article, to understand its key findings and relevance to their information needs.

Although Lay Summaries are required or encouraged by some biomedical publications, they are not universal, leaving a significant amount of inaccessible to lay audiences. Furthermore, the burden of writing these summaries is often placed upon the article authors, who are not always adept at effectively communicating their work to a non-technical audience. As such, automatic lay summary generation has significant potential to help both authors and non-expert readers by improving the dissemination of important research.

The BioLaySumm shared task[1] focuses on improving the automatic lay summarization of biomedical research. Through this shared task, we aim to encourage the development of novel approaches and increase interest in the research of automatic techniques for disseminating scientific research to broad audiences. In this paper, we present the results of the second edition of the BioLaySumm shared task, hosted by the BioNLP Workshop at ACL 2024.

In what remains of the paper, we address the task formulation (§2), datasets (§3), and evaluation procedure (§4), before providing a description of overall results and notable insights (§5), and finall the participating systems (§6).

2 Task Description

As part of the task, participants must develop systems capable of generating a lay summary of biomedical research, given the article's text as input. Our competition was hosted using the CodaBench platform (Xu et al., 2022).

As with the previous edition of the task, two separate datasets, **PLOS** and **eLife** are used. Participants were provided with both training and valida-

[1] https://biolaysumm.org

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 122–131
August 16, 2024. ©2024 Association for Computational Linguistics

tion sets, complete with reference lay summaries, alongside a blind test set. Final system performance is determined by the performance of participants' systems on the blind test set, which could be obtained by submitting their predicted lay summaries to our CodaBench competition, where they were automatically evaluated. More information regarding our datasets and evaluation protocol is provided in subsequent sections (§3 and §4, respectively).

We allowed submissions to be generated from either two separate summarisation models (i.e., one trained on each dataset) or a single unified model (i.e., trained on both datasets). Participants were required to indicate which approach was taken for each submission, in addition to whether or not they made use of additional training data (i.e., data not provided specifically for the task). Participants were also allowed to compete as part of teams, where each team was permitted to make a maximum of 15 test set submissions to the CodaBench platform.[2] However, teams were required to select only one of their submissions to appear on the final task leaderboard.

Because of its strong performance in the previous edition of the task, we also choose to keep the same baseline system. Specifically, this baseline system consists of two separate BART-base models (Lewis et al., 2020), trained independently on the PLOS and eLife datasets.

3 Datasets

The datasets used for the task are based on the previous works of Goldsack et al. (2022) and Luo et al. (2022b), and are derived from two different biomedical publications: **Public Library of Science (PLOS)** and **eLife**. Each dataset consists of biomedical research articles paired with expert-written lay summaries.

As described in Goldsack et al. (2022), the lay summaries of each dataset also exhibit numerous notable differences in their characteristics, with eLife's lay summaries being longer, more abstractive, and more readable than those of PLOS.

Furthermore, for PLOS, lay summaries are author-written, and articles are derived from 5 peer-reviewed journals covering Biology, Computational Biology, Genetics, Pathogens, and Neglected Tropical Diseases. For eLife, lay summaries are written by expert editors (in correspondence with authors), and articles are derived from the peer-reviewed eLife journal, covering all areas of the life sciences and medicine. For a more detailed analysis of dataset content, readers can refer to Goldsack et al. (2022).

Dataset	Subtask	# Train	# Val	# Test
eLife	1	4,346	241	142
PLOS	1, 2	24,773	1,376	142*

Table 1: Data split sizes for each dataset. * denotes that this split is different for each subtask.

Table 1 summarises the data split information for both datasets. Note that the training and validation sets used for both datasets are equal to those published in Goldsack et al. (2022).

As with the previous task edition, we collect new test splits for both PLOS and eLife data using more recently published articles from each respective journal. This test data consists of 142 PLOS articles and 142 eLife articles.

In utilizing these datasets for our task, we hope to enable the training of abstractive summarisation models that are capable of generating lay summaries for unseen articles covering a wide range of biomedical topics, enabling the significance of new, important publications to be effectively communicated to non-expert audiences.

4 Evaluation

For both subtasks, we evaluate summary quality according to three criteria - *Relevance*, *Readability*, and *Factuality* - where each criterion is composed of one or more automatic metrics:

- *Relevance*: ROUGE-1, 2, and L (Lin, 2004) and BERTScore (Zhang et al., 2020b).
- *Readability*: Flesch-Kincaid Grade Level (FKGL), Dale-Chall Readability Score (DCRS), *Coleman-Laiu Index (CLI), and *LENS (Maddela et al., 2023).
- *Factuality*: *AlignScore (Zha et al., 2023) and *SummaC (Zha et al., 2023)

Where "*" indicates that the metric is newly introduced for this year's edition of the task. Specifically, the CLI and LENS metrics are introduced in order to enhance our evaluation of summary readability. Alternatively, AlignScore and SummaC are introduced to replace the fine-tuned BARTScore

[2] A significant increase on the limit of 3 submissions imposed in the previous edition of the task.

model used to assess factuality in the previous task edition, with the reason for this being that BARTScore was found to exhibit bias toward BART-based approaches.

The scores calculated for each metric are the average of those calculated independently for the generated lay summaries of PLOS and eLife. The aim is to maximize the scores for all metrics except for FKGL, DCRS, and CLI the Readability metrics. For these metrics, the aim is to minimize scores, as lower scores are indicative of greater readability.[3]

Following the submission deadline for each subtask, an overall ranking is calculated based on the average performance of submissions across all criteria. Specifically, we first apply min-max normalization to the scores of each metric (thus establishing a common value range), before averaging across metrics within each criterion to obtain criterion-level scores.[4] Note that, for metrics that we minimize (i.e., FKGL, DCRS, and CLI) we calculate 1 minus the mix-max normalized value. Finally, criterion-level scores are then averaged to obtain an overall score, by which submissions are then ranked.

5 Task Results

Table 2 presents the performance of the submission selected by each team to appear on the final leaderboard, according to the defined task metrics.

Overall, the final leaderboard of BioLaySumm 2024 contains a total of 53 teams, who made a combined total of over 200 submissions. This represents a 165% increase in participation over BioLaySumm 2023, which attracted a total of 20 teams across two subtasks. In this section, we summarize some of the key results and notable trends that were observed among participants.

Model selection trends We identify several trends amongst participants in terms of the models used for experiments.[5] Firstly, the use of Large Language Models was found to be particularly prevalent, with a total of 18 teams indicating that LLMs were used in some capacity. Compared to the 3 teams who used LLMs in BioLaySumm 2023, this represents a stark increase that is reflective of shifts in the broader research landscape of NLP. Within those teams using LLMs, biomedical-specific models such as BioGPT (Luo et al., 2022a) and BioMistral (Labrak et al., 2024) proved popular, with 7 teams indicating they used such models. Other LLMs used include GPT-4 (2), LLAMA (2), and Claude (1). There is evidence that LLMs were used for both summary generation and summary post-processing, with various settings (including fine-tuned, few-shot, and zero-shot) being adopted.

Outside of LLMs, the T5 (Raffel et al., 2019) model family proved the most popular alternative approach, with 13 teams making use of these models in their selected submissions. In particular, the FLAN-T5 (Chung et al., 2022a) model was found to be widely-used, being selected by 9 teams. Interestingly, only 3 teams were found to have used BART-based models, a significant drop from the previous BioLaySumm edition, where they were the most widely adopted approach. We find this shift in model selection to be an encouraging sign that participants are keen to explore novel methods for Lay Summarisation, in line with our overall task objectives.

Baseline comparison As shown in Table 2, 5 teams exceed the overall rank of the BART baseline system. This represents an an increase on the previous edition of the task, whereby only 1 team outperformed the same baseline system in terms of overall ranking.

Number of models used Contrary to the previous task edition, we find that more teams opted for the use of a single unified model for both datasets (27 out of 53), as opposed to using one model for each dataset. This is likely a result of a significant increase in the use of Large Language Models, an unsurprising shift that reflects the current research landscape in Natural Language Processing. Interestingly, the top four ranked teams can all be seen to adopt a 2-model approach, indicative of the potential benefits of having a distinct model specifically catering to the different lay summary styles of each dataset.

Use of additional data As with this previous task edition, we found that very few teams opted to make use of additional data (i.e., data not provided by the organizers as part of the task) in model development. As shown by the **+** column in Table 2,

[3] For these metrics, the scores are estimates of the US Grade level of education required to comprehend a given text.

[4] This represents a minor change from the averaging protocol of the previous task edition, in which we first calculated rankings for each criterion, before summing these rankings to compute an overall rank.

[5] Information about model selection is derived from both system papers submitted by participants and a “system description" field included in the submission form on CodaBench.

⋆	Team	#	+	Relevance				Readability				Factuality	
				R-1	**R-2**	**R-L**	**BertS**	**FKGL**	**DCRS**	**CLI**	**LENS**	**AlignS**	**SummaC**
1	**UIUC_BioNLP**	2	×	48.55	**15.69**	**45.50**	**86.77**	11.75	9.34	13.36	52.85	80.04	73.38
2	**Ctyun AI**	2	×	47.96	15.46	44.94	86.66	12.44	9.67	14.15	51.09	82.72	74.80
3	**Saama Technologies**	2	×	47.85	15.45	44.97	86.70	11.36	9.10	13.15	51.90	77.83	72.68
4	**WisPerMed**	2	×	47.12	15.18	44.28	86.53	11.07	8.86	12.87	51.03	78.18	72.16
5	**cylaun**	1	×	47.39	14.55	44.45	85.61	**10.46**	9.33	12.64	41.69	75.26	78.44
6	**BART Baseline**	2	×	46.96	13.95	43.58	86.23	12.04	10.15	13.49	48.10	77.88	70.26
7	**AUTH**	1	✓	48.23	14.57	44.77	85.76	12.44	10.04	13.50	66.11	74.18	66.40
8	**maverick**	1	×	42.77	12.97	39.42	85.01	15.04	10.65	16.61	52.30	91.22	83.85
9	**Empress**	1	×	43.96	12.29	41.36	84.89	10.66	9.06	12.89	59.73	73.47	68.02
10	**eulerian**	1	×	40.35	11.66	37.10	84.51	14.80	10.76	16.53	48.46	91.73	85.38
11	**BioLay_AK_SS**	2	×	43.98	12.15	40.39	84.71	14.20	11.12	15.12	49.57	85.03	78.60
12	**HULAT-UC3M**	2	×	**48.72**	14.65	45.20	86.22	12.71	10.43	14.08	49.34	66.69	67.03
13	**Atif_Tanish**	1	×	43.82	11.96	41.01	84.84	10.61	9.12	12.86	60.14	72.92	67.12
14	**qwerty**	1	×	37.26	10.45	34.48	83.54	13.36	9.18	14.60	42.16	89.89	83.23
15	**Deakin**	2	×	48.22	14.20	44.41	85.83	14.46	10.76	15.48	63.91	74.57	61.80
16	**MDSCL**	2	×	42.56	13.01	39.35	85.20	14.01	10.78	15.92	63.05	81.50	71.54
17	**MDS-CL**	2	×	42.13	12.90	38.93	85.14	14.13	10.82	15.96	61.71	81.98	73.14
18	**elirf**	2	✓	48.15	13.66	43.09	85.95	13.61	10.86	14.66	48.02	78.21	60.66
19	**RAG-RLRC-LaySum**	2	×	46.24	13.04	42.37	85.29	12.68	10.43	14.41	59.26	71.28	66.29
20	**naive_bhais**	2	×	43.42	12.60	39.91	85.72	12.89	10.94	14.32	37.86	81.34	67.81
21	**MDS-CL**	1	×	42.31	11.05	39.22	85.62	11.93	9.23	13.25	74.67	71.52	56.55
22	**MDS-CL**	1	×	43.43	11.98	40.13	85.55	12.39	9.76	14.28	76.80	72.18	54.41
23	**DhruvShlo**	1	×	42.15	11.05	39.40	84.42	11.76	9.08	13.02	49.17	71.25	63.98
24	**naman_tejas**	1	×	39.54	11.06	36.73	84.25	12.29	9.20	13.58	50.44	75.68	68.10
25	**SINAI**	2	×	42.05	12.49	38.53	85.83	12.23	9.86	13.81	76.95	71.17	53.98
26	**XYZ**	2	×	41.04	9.93	38.01	85.50	11.02	9.37	13.00	**81.21**	70.18	54.63
27	**gpsigh**	2	×	33.60	9.18	30.97	82.97	15.69	9.30	15.17	42.06	91.28	82.11
28	**YXZ**	2	×	42.25	10.91	39.20	84.99	11.18	8.57	12.44	71.57	64.89	53.49
29	**sanika**	2	×	42.90	11.16	38.06	83.33	17.93	12.40	17.37	11.37	85.16	**90.28**
30	**Bossy Beaver**	1	×	41.32	11.45	37.98	84.73	13.99	10.41	15.74	65.49	78.08	60.65
31	**Dayal K-Laksh G**	1	×	33.93	9.49	30.55	84.98	14.39	12.15	16.24	32.33	93.07	80.71
32	**MKGS**	1	×	37.75	9.72	34.67	83.33	15.79	11.92	17.50	22.07	**93.08**	83.52
33	**Shallow-Learning**	1	×	42.22	11.33	39.54	83.89	10.56	9.04	12.42	53.68	57.28	61.17
34	**NLPSucks**	2	×	34.91	8.32	33.32	82.62	10.68	**6.76**	12.08	37.86	74.36	64.22
35	**CookieMonster**	2	×	43.06	10.33	39.83	84.57	12.04	9.37	13.18	49.53	63.63	59.19
36	**NoblesseUranium**	1	×	39.16	10.34	35.87	84.65	14.21	10.44	15.45	51.99	75.74	67.32
37	**roon**	2	×	44.16	11.24	41.44	84.74	11.78	8.86	12.38	71.26	52.73	50.50
38	**jimmyapples**	2	×	43.36	10.84	40.35	84.82	11.44	9.03	12.10	71.48	56.54	49.00
39	**Shivam**	2	×	33.85	8.91	30.76	83.56	12.90	11.89	15.14	15.66	91.64	80.94
40	**HGP_NLP**	2	×	29.69	8.60	26.95	83.74	11.20	9.91	12.79	44.22	79.45	74.11
41	**Cornell-BioLay**	1	×	39.50	7.92	35.99	84.50	10.97	9.56	12.66	72.53	60.10	51.76
42	**xpc**	2	×	44.59	11.80	40.36	84.84	13.45	10.33	15.72	67.72	56.87	48.78
43	**Hemlo**	1	×	30.04	6.88	27.86	81.10	16.49	7.58	15.74	21.91	89.50	74.41
44	**anjaneya**	2	×	28.87	8.26	26.00	83.66	13.70	11.45	15.46	37.03	74.71	78.66
45	**Runtime_Terror**	1	×	40.18	10.14	37.44	83.69	13.98	8.41	13.13	49.60	47.51	50.37
46	**Abhi_Sidd**	1	×	35.33	9.15	31.79	83.09	17.29	12.42	14.47	17.41	79.09	62.95
47	**cbdch**	1	×	36.69	9.29	32.98	85.09	14.43	11.18	15.82	74.13	63.34	44.61
48	**aLoneLM**	1	×	35.67	6.74	33.29	82.33	11.07	8.52	**11.04**	42.82	48.93	52.62
49	**hohoho**	2	×	33.46	5.48	31.32	81.93	11.21	8.80	12.04	53.01	44.68	50.98
50	**huizige**	1	×	37.16	8.82	33.59	83.06	15.40	11.39	16.78	47.53	60.13	49.36
51	**SSS**	1	×	25.64	6.21	23.18	82.81	13.71	12.45	16.61	43.82	72.72	61.55
52	**H2P**	1	×	25.93	4.03	23.73	81.70	16.53	11.94	18.98	56.77	56.03	47.04
53	**KnowLab**	1	✓	32.16	7.34	28.31	80.63	36.29	11.55	11.28	1.32	42.41	54.74

Table 2: Task leaderboard - all metrics. The ⋆ column denotes the submission rank, the **#** column the number of models used - 1 (unified) or 2 (one for each dataset), and the **+** column the use of additional training data. **R** = ROUGE F1, **BertS** = BertScore, **FKGL** = Flesch-Kincaid Grade Level, **DCRS** = Dale-Chall Readability Score, **CLI** = Coleman-Liau Index, **AlignS** = AlignScore.

Rank	Team	Relevance				Readability				Factuality	
		R-1	**R-2**	**R-L**	**BertS**	**FKGL**	**DCRS**	**CLI**	**LENS**	**AlignS**	**SummaC**
1	**UIUC_BioNLP**	0.993	**1.000**	**1.000**	**1.000**	0.950	0.547	0.708	0.645	0.743	0.630
2	**Ctyun AI**	0.967	0.980	0.975	0.982	0.923	0.488	0.609	0.623	0.796	0.661
3	**Saama Technologies**	0.962	0.979	0.976	0.989	0.965	0.589	0.734	0.633	0.699	0.615
4	**WisPerMed**	0.931	0.956	0.945	0.962	0.976	0.631	0.770	0.622	0.706	0.603
5	**cylaun**	0.943	0.902	0.953	0.811	**1.000**	0.548	0.800	0.505	0.648	0.741
6	**BART Baseline**	0.924	0.851	0.914	0.913	0.939	0.405	0.693	0.586	0.700	0.561
7	**AUTH**	0.979	0.904	0.967	0.836	0.923	0.424	0.691	0.811	0.627	0.477
8	**maverick**	0.742	0.766	0.728	0.713	0.822	0.316	0.298	0.638	0.963	0.859
9	**Empress**	0.794	0.708	0.815	0.695	0.992	0.596	0.768	0.731	0.613	0.513
10	**eulerian**	0.637	0.655	0.624	0.632	0.832	0.298	0.308	0.590	0.973	0.893
11	**BioLay_AK_SS**	0.795	0.697	0.771	0.664	0.855	0.233	0.486	0.604	0.841	0.744
12	**HULAT-UC3M**	**1.000**	0.911	0.987	0.912	0.913	0.355	0.618	0.601	0.479	0.491
13	**Atif_Tanish**	0.788	0.680	0.799	0.686	0.994	0.585	0.771	0.736	0.602	0.493
14	**qwerty**	0.503	0.551	0.506	0.475	0.888	0.574	0.552	0.511	0.937	0.845
15	**Deakin**	0.978	0.872	0.951	0.848	0.845	0.298	0.441	0.784	0.635	0.376
16	**MDSCL**	0.733	0.770	0.725	0.745	0.862	0.294	0.385	0.773	0.771	0.590
17	**MDS-CL**	0.714	0.761	0.706	0.734	0.858	0.286	0.381	0.756	0.781	0.625
18	**elirf**	0.976	0.826	0.892	0.867	0.878	0.280	0.544	0.585	0.707	0.351
19	**RAG-RLRC-LaySum**	0.892	0.772	0.860	0.759	0.914	0.355	0.576	0.725	0.570	0.475
20	**naive_bhais**	0.790	0.749	0.773	0.850	0.912	0.270	0.578	0.498	0.760	0.493
21	**MDS-CL**	0.722	0.601	0.719	0.813	0.943	0.567	0.722	0.918	0.575	0.261
22	**MDS-CL**	0.771	0.682	0.759	0.802	0.925	0.473	0.592	0.945	0.588	0.214
23	**DhruvShlo**	0.716	0.602	0.727	0.618	0.950	0.593	0.751	0.599	0.569	0.424
24	**naman_tejas**	0.602	0.603	0.607	0.589	0.929	0.571	0.681	0.615	0.657	0.514
25	**SINAI**	0.711	0.726	0.688	0.848	0.932	0.455	0.651	0.947	0.568	0.205
26	**XYZ**	0.667	0.506	0.665	0.793	0.978	0.542	0.754	**1.000**	0.548	0.219
27	**gpsigh**	0.345	0.442	0.349	0.381	0.798	0.553	0.481	0.510	0.965	0.821
28	**YXZ**	0.720	0.590	0.718	0.711	0.972	0.682	0.825	0.879	0.444	0.194
29	**sanika**	0.748	0.612	0.667	0.440	0.711	0.008	0.203	0.126	0.844	**1.000**
30	**Bossy Beaver**	0.679	0.636	0.663	0.668	0.863	0.359	0.409	0.803	0.704	0.351
31	**Dayal K-Laksh G**	0.359	0.468	0.330	0.708	0.848	0.053	0.346	0.388	**1.000**	0.790
32	**MKGS**	0.525	0.488	0.515	0.440	0.794	0.093	0.187	0.260	**1.000**	0.852
33	**Shallow-Learning**	0.718	0.626	0.733	0.531	0.996	0.600	0.826	0.655	0.293	0.362
34	**NLPSucks**	0.402	0.368	0.454	0.324	0.991	**1.000**	0.870	0.457	0.631	0.429
35	**CookieMonster**	0.755	0.540	0.746	0.643	0.939	0.541	0.731	0.604	0.419	0.319
36	**NoblesseUranium**	0.586	0.541	0.568	0.656	0.855	0.353	0.445	0.634	0.658	0.497
37	**roon**	0.802	0.618	0.818	0.670	0.949	0.632	0.832	0.875	0.204	0.129
38	**jimmyapples**	0.768	0.584	0.769	0.683	0.962	0.601	0.867	0.878	0.279	0.096
39	**Shivam**	0.356	0.418	0.340	0.477	0.905	0.098	0.484	0.179	0.972	0.795
40	**HGP_NLP**	0.175	0.392	0.169	0.507	0.971	0.446	0.781	0.537	0.731	0.646
41	**Cornell-BioLay**	0.601	0.333	0.574	0.630	0.980	0.508	0.797	0.891	0.349	0.156
42	**xpc**	0.821	0.666	0.770	0.686	0.884	0.372	0.411	0.831	0.285	0.091
43	**Hemlo**	0.190	0.245	0.210	0.076	0.766	0.857	0.408	0.258	0.929	0.652
44	**anjaneya**	0.140	0.363	0.126	0.493	0.874	0.176	0.444	0.447	0.637	0.745
45	**Runtime_Terror**	0.630	0.524	0.639	0.499	0.864	0.711	0.737	0.604	0.101	0.126
46	**Abhi_Sidd**	0.420	0.439	0.386	0.400	0.735	0.005	0.568	0.201	0.724	0.401
47	**cbdch**	0.479	0.451	0.439	0.726	0.846	0.224	0.399	0.911	0.413	0.000
48	**aLoneLM**	0.435	0.233	0.453	0.277	0.976	0.691	**1.000**	0.520	0.129	0.175
49	**hohoho**	0.339	0.124	0.365	0.212	0.971	0.642	0.874	0.647	0.045	0.139
50	**huizige**	0.499	0.410	0.466	0.395	0.809	0.187	0.277	0.578	0.350	0.104
51	**SSS**	0.000	0.187	0.000	0.356	0.874	0.000	0.299	0.532	0.598	0.371
52	**H2P**	0.013	0.000	0.025	0.174	0.765	0.089	0.000	0.694	0.269	0.053
53	**KnowLab**	0.283	0.283	0.230	0.000	0.000	0.158	0.971	0.000	0.000	0.222

Table 3: Task leaderboard with min-max normalization. **R** = ROUGE F1, **BertS** = BertScore, **FKGL** = Flesch-Kincaid Grade Level, **DCRS** = Dale-Chall Readability Score, **CLI** = Coleman-Liau Index, **AlignS** = AlignScore.

only three teams - **AUTH**, **elirf**, and **KnowLab** - indicated that they adopted such an approach.

Reflection on evaluation protocol changes Here, we discuss the impact of the changes made to the evaluation protocol over the previous task edition. As mentioned in §4, the first of these changes surrounds the introduction of new metrics for the Readability and Factuality criteria. As a model-based simplification metric, LENS was introduced to provide an additional angle for teams to consider for Readability, with Maddela et al. (2023) demonstrating that the metric correlates particularly well with the *fluency* ratings of human annotators for simplified texts. Notably, LENS does not exhibit a strong alignment with other (more heuristic) Readability metrics, suggesting that these metrics may not capture this aspect of simplified texts.

For Factuality, we introduced the AlignScore and SummaC metrics as a replacement for a fine-tuned version of BARTScore to avoid potential bias toward BART-based models. However, given that these metrics broadly involve comparing a generated summary to the source text, these metrics tend to favor highly extractive outputs. Given that reference lay summaries tend to be quite abstractive (particularly in the case of the eLife dataset), this resulted in a trade-off between scoring highly for Factuality and the metrics of Relevance or Readability. Overall, we observe that the systems that ranked the highest were those that most successfully balanced this trade-off, typically obtaining strong Relevance and Readability scores while maintaining relatively high Factuality scores.

Finally, the process for the calculation of final rankings was changed from summing individual criterion rankings to the averaging of average criterion scores. This change was motivated by the failure of the previous method of ranking to take into account the relative difference between average scores for a given criterion, something that was commented on by last year's participants.[6] However, the new ranking system was also found to be not without its issues, particularly surrounding the existence of outliers. Specifically, it was observed that, if there existed teams that scored particularly poorly for a given metric, then all other teams would obtain relatively strong (and less diverse) scores for this metric relative to others - this can be seen for the FKGL metric in Table 3.

6 Submissions

Out of the 53 participating teams, 14 teams submitted system papers. Here, we provide a brief summary of the approaches taken by these teams.

UIUC_BioNLP (You et al., 2024) This team produced the top-ranked submission, adopting an extract-then-summarize approach that utilizes TextRank (Mihalcea and Tarau, 2004) for salient sentence extraction, followed by a fine-tuned GPT-3.5-turbo model for summary generation. Specifically, their submitted system extracted the top 40 most salient sentences using TextRank, and their GPT-bsaed model is fine-tuned on 200 examples. Additional experimentation was conducted using various extractive summarization approaches and comparing the number of examples required for effective fine-tuning. Furthermore, the team also explored a LongFormer-based approach that further incorporates retrieved Wikipedia data in a Retrieval-Augmented Generation (RAG) setup.

Cytun AI (Zhao et al., 2024) Making the second-ranked submission, the methodology of this team surrounds the use of fine-tuned LLMs. As part of their experimentation, they compare two approaches for handling lengthy input articles: hard truncation and text chunking. Additionally, their summary-generation pipeline includes data preprocessing, augmentation, and prompt engineering.

Saama Technologies (Kim et al., 2024) This team achieved the third-ranking submission, which surrounded fine-tuning a Mistral-7B model[7] in an unsupervised fashion using low-ranked adaptation (LoRA) (Hu et al., 2021), followed by zero-shot summary generation and post-processing to remove redundant sentences. This team also experiments with several other fine-tuning methods, including supervised fine-tuning with LoRA and Direct Preference Optimization (Rafailov et al., 2023).

WisPerMed (Pakull et al., 2024) Ranking in fourth place, the selected submission of WisPerMed utilized a fine-tuned BioMistral model, combined with few-shot prompting and a Dynamic-Expert selection (DES) mechanism. Specifically, their BioMistral Model was trained using abstracts and lay summaries of the provided train set; and

[6] For example, in terms of average criterion score, the team ranked 1st may outperform the team ranked 2nd by a large margin, who in term may outperform the team ranked 3rd by a small margin. However, by converting these scores to rankings, all differences are treated as equal.

[7] `mistral-7B-instruct-v0.2`

their proposed DES mechanism involved generating several lay summary versions with different prompts for a given input, before selecting the most desirable based on the scores of the references-less Readability and Factuality metrics used in the task. In additional experiments, they also measured system performance utilizing LLAMA3, as well as that of few-shot and zero-shot model variants. The task organizers selected this team to receive an award for the “most innovative approach".

AUTH (Stefanou et al., 2024) Being one of the only teams to utilize external data, this retrieves 300 abstract-lay abstract pairs scraped from the Science Journal for Kids website.[8] They use this retrieved data as in-context examples for GPT-4, which they prompt to augment the provided datasets by rewriting reference summaries with higher readability scores. Finally, they use this data to fine-tune to fine-tune BioBART (Yuan et al., 2022), whilst also experimenting with controllable generation techniques in the form of control tokens prepended to the input article (`<elife>` / `<plos>` and `<general_lay_summary>` / `<kids_lay_summary>`).

Eulerian (Modi and Karthikeyan, 2024) The team experimented with different combinations of the FLAN-T5 (Chung et al., 2022b) model variations and data selection. They compare the performance of these methods with a preposed “Preprocessing over Abstract" technique, in which they use a regular expression to remove some abstract information (i.e., anything inside of parentheses, braces and brackets), finding that this outperforms all neural methods tested in terms of Relevance and Factuality metrics.

BioLay_AK_SS (Karotia and Susan, 2024) Focusing largely on data augmentation, this team generated additional summary samples using 2 general-purpose models: BART (Lewis et al., 2020) and PEGASUS (Zhang et al., 2020a). The augmented dataset was then used to fine-tune a domain-specific BioBART model, which was found to improve its overall improved overall performance.

HULAT-UC3M (Gonzalez and Martínez, 2024) Again comparing the performance of domains-specific and general-purpose models, this team experimented with fine-tuning both Longformer (Beltagy et al., 2020) and BioBART models on the given datasets. Additionally, they experiment with extending BioBART to utilise Longformer-based sparse attention, thus allowing it to process longer inputs. Overall, they found that fine-tuning the standard BioBART model on each dataset yields the best performance.

DeakinNLP (Quoc To et al., 2024) This team assessed the performance of both a fine-tuned Longformer and GPT-4 (with zero- and few-shot prompting). Additional analysis is also conducted surrounding data selection and the performance vs. cost trade-off between select methods.

elirf (Ahuir et al., 2024) Again utilising Longformer as their base model, this team experimented with domain-adaption via a continuous pre-training approach. During pre-training, several pretraining tasks were aggregated to inject linguistic knowledge and increase the abstractiveness of generated summaries. Finally, they developed a regression-based ranking model that improved system performance by selecting the most promising from a set of generated summaries.

RAG-RLRC-LaySum (Ji et al., 2024) This team developed a Retrieval-Augmented Generation (RAG) Lay Summarisation approach, utilizing multiple knowledge sources (including both source documents and Wikipedia). They experiment with LLMs (Gemini and ChatGPT) for both summary generation and paraphrasing, in addition to a Longformer baseline. Lastly, the team also develop a Reinforcement Learning strategy to fine-tune the readability of generated summaries.

SINAI (Chizhikova et al., 2024) Focusing largely on a few-shot setting, this team compared the performance of several popular LLMs including GPT-3.5, GPT-4, and LLAMA3. Further experimentation surrounded the fine-tuning of a smaller LLAMA model (LLAMA3-8B) using both parameter-efficient LoRA techniques and Direct Preference Optimization (Rafailov et al., 2023).

XYZ (Zhou et al., 2024) This team performed a thorough comparison of several state-of-the-art LLMs, focusing largely on comparing the readability of generated summaries. Further experimentation surrounds Summary rewriting, Title infusing, K-shot prompting, and LoRA-based fine-tuning, with their best-performing submission utilizing a combination of these methods and obtaining the best overall Readability scores.

[8] https://sciencejournalforkids.org/

HGP_NLP (Malik et al., 2024) This team fine-tune and evaluate multiple T5 model variants, also experimenting with LoRA-based fine-tuning.

7 Conclusion

The second edition of the BioLaySumm Shared Task was hosted by the BioNLP Workshop@ACL 2024. Several changes were implemented over the previous edition of the task covering participation rules, evaluation metrics, and ranking protocol. In terms of participant engagement, the task attracted a total of 53 teams, representing a significant growth from the previous edition's 20 teams. Our results indicate a drastic shift towards the use of LLMs for lay summarisation, with a wide range of both domain-specific and general-purpose LLMs being adopted in various settings across participant submissions.

References

Vicent Ahuir, Diego Torres, Encarna Segarra, and Lluís-F Encarna. 2024. Elirf-vrain at biolaysumm: Boosting lay summarization systems performance with ranking models. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv:2004.05150*.

Mariia Chizhikova, Manuel Carlos Díaz-Galiano, L. Alfonso Ureña-López, and María-Teresa Martín-Valdivia. 2024. Sinai at biolaysumm: Self-play finetuning of large language models for biomedical lay summarisation. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, S. Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Wei Yu, Vincent Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed Huai hsin Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022a. Scaling instruction-finetuned language models. *ArXiv*, abs/2210.11416.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022b. Scaling instruction-finetuned language models.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Adrian Gonzalez and Paloma Martínez. 2024. Hulat-uc3m at biolaysumm: Adaptation of biobart and longformer models to summarizing biomedical documents. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685.

Yuelyu Ji, Zhuochun Li, Rui Meng, Sonish Sivarajkumar, Yanshan Wang, Zeshui Yu, Hui Ji, Yushui Han, Hanyu Zeng, and Daqing He. 2024. Rag-rlrc-laysum at biolaysumm: Integrating retrieval-augmented generation and readability control for layman summarization of biomedical texts. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Akanksha Karotia and Seba Susan. 2024. Biolay_ak_ss at biolaysumm: Domain adaptation by two-stage finetuning of large language models used for biomedical lay summary generation. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Hwanmun Kim, Kamal raj Kanakarajan, and Malaikannan Sankarasubbu. 2024. Saama technologies at biolaysumm: Abstract based fine-tuned models with lora. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. Biomistral: A collection of open-source pretrained large language models for medical domains.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. 2022a. Biogpt: Generative pre-trained transformer for biomedical text generation and mining. *Briefings in bioinformatics*.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2022b. Readability controllable biomedical document summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4667–4680, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. LENS: A learnable evaluation metric for text simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16383–16408, Toronto, Canada. Association for Computational Linguistics.

Hemang Malik, Gaurav Pradeep, and Pratinav Seth. 2024. Hgp-nlp at biolaysumm: Leveraging lora for lay summarization of biomedical research articles using seq2seq transformers. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Satyam Modi and T Karthikeyan. 2024. Eulerian at biolaysumm: Preprocessing over abstract is all you need. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Tabea Margareta Grace Pakull, Hendrik Damm, Ahmad Idrissi-Yaghir, Henning Schäfer, Peter A. Horn, and Christoph M. Friedrich. 2024. Wispermed at biolaysumm: Adapting autoregressive large language models for lay summarization of scientific articles. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Huy Quoc To, Ming Liu, and Guangyan Huang. 2024. Deakinnlp at biolaysumm: Evaluating fine-tuning longformer and gpt-4 prompting for biomedical lay summarization. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Loukritia Stefanou, Tatiana Passali, and Grigorios Tsoumakas. 2024. Auth at biolaysumm 2024: Bringing scientific content to kids. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Zhen Xu, Sergio Escalera, Adrien Pavão, Magali Richard, Wei-Wei Tu, Quanming Yao, Huan Zhao, and Isabelle Guyon. 2022. Codabench: Flexible, easy-to-use, and reproducible meta-benchmark platform. *Patterns*, 3(7):100543.

Zhiwen You, Shruthan Radhakrishna, Shufan Ming, and Halil Kilicoglu. 2024. Uiuc_bionlp at biolaysumm: an extract-then-summarize approach augmented with wikipedia knowledge for biomedical lay summarization. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Hongyi Yuan, Zheng Yuan, Ruyi Gan, Jiaxing Zhang, Yutao Xie, and Sheng Yu. 2022. BioBART: Pretraining and evaluation of a biomedical generative language model. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 97–109, Dublin, Ireland. Association for Computational Linguistics.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020a. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Ruijing Zhao, Siyu Bao, Siqin Zhang, Jinghui Zhang, Weiyin Wang, and Yunian Ru. 2024. Ctyun ai at biolaysumm: Enhancing lay summaries of biomedical articles through large language models and data augmentation. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

Jieli Zhou, Cheng Ye, Pengcheng Xu, and Hongyi Xin. 2024. Team yxz at biolaysumm: Adapting large language models for biomedical lay summarization. In *Proceedings of the 23rd Workshop on Biomedical Language Processing*, Bangkok, Thailand. Association for Computational Linguistics.

A Appendix

⋆	Team	Relevance	Readability	Factuality	Avg.
1	**UIUC_BioNLP**	0.998	0.712	0.686	0.799
2	**Ctyun AI**	0.976	0.661	0.728	0.788
3	**Saama Technologies**	0.977	0.730	0.657	0.788
4	**WisPerMed**	0.948	0.750	0.655	0.784
5	**cylaun**	0.902	0.713	0.694	0.770
6	**BART Baseline**	0.901	0.655	0.631	0.729
7	**AUTH**	0.922	0.713	0.552	0.729
8	**maverick**	0.737	0.519	0.911	0.723
9	**Empress**	0.753	0.772	0.563	0.696
10	**eulerian**	0.637	0.507	0.933	0.692
11	**BioLay_AK_SS**	0.732	0.545	0.793	0.690
12	**HULAT-UC3M**	0.952	0.622	0.485	0.686
13	**Atif_Tanish**	0.738	0.772	0.547	0.686
14	**qwerty**	0.509	0.631	0.891	0.677
15	**Deakin**	0.912	0.592	0.506	0.670
16	**MDSCL**	0.743	0.579	0.681	0.667
17	**MDS-CL**	0.729	0.570	0.703	0.667
18	**elirf**	0.890	0.572	0.529	0.664
19	**RAG-RLRC-LaySum**	0.821	0.643	0.522	0.662
20	**naive_bhais**	0.790	0.565	0.626	0.660
21	**MDS-CL**	0.714	0.788	0.418	0.640
22	**MDS-CL**	0.753	0.734	0.401	0.629
23	**DhruvShlo**	0.666	0.723	0.497	0.628
24	**naman_tejas**	0.600	0.699	0.585	0.628
25	**SINAI**	0.743	0.746	0.386	0.625
26	**XYZ**	0.658	0.819	0.384	0.620
27	**gpsigh**	0.379	0.585	0.893	0.619
28	**YXZ**	0.685	0.840	0.319	0.614
29	**sanika**	0.617	0.262	0.922	0.600
30	**Bossy Beaver**	0.662	0.609	0.528	0.599
31	**Dayal K-Laksh G**	0.466	0.409	0.895	0.590
32	**MKGS**	0.492	0.333	0.926	0.584
33	**Shallow-Learning**	0.652	0.769	0.328	0.583
34	**NLPSucks**	0.387	0.830	0.530	0.582
35	**CookieMonster**	0.671	0.704	0.369	0.581
36	**NoblesseUranium**	0.588	0.572	0.577	0.579
37	**roon**	0.727	0.822	0.166	0.572
38	**jimmyapples**	0.701	0.827	0.187	0.572
39	**Shivam**	0.398	0.417	0.884	0.566
40	**HGP_NLP**	0.311	0.684	0.688	0.561
41	**Cornell-BioLay**	0.535	0.794	0.253	0.527
42	**xpc**	0.736	0.625	0.188	0.516
43	**Hemlo**	0.180	0.572	0.791	0.514
44	**anjaneya**	0.281	0.485	0.691	0.486
45	**Runtime_Terror**	0.573	0.729	0.113	0.472
46	**Abhi_Sidd**	0.411	0.378	0.563	0.450
47	**cbdch**	0.524	0.595	0.207	0.442
48	**aLoneLM**	0.349	0.797	0.152	0.433
49	**hohoho**	0.260	0.783	0.092	0.378
50	**huizige**	0.443	0.463	0.227	0.378
51	**SSS**	0.136	0.426	0.484	0.349
52	**H2P**	0.053	0.387	0.161	0.200
53	**KnowLab**	0.199	0.282	0.111	0.197

Table 4: Task leaderboard with min-max normalization. The ⋆ column denotes the submission rank. **R** = ROUGE F1, **BertS** = BertScore, **FKGL** = Flesch-Kincaid Grade Level, **DCRS** = Dale-Chall Readability Score, **CLI** = Coleman-Liau Index, **AlignS** = AlignScore.

UIUC_BioNLP at BioLaySumm: An Extract-then-Summarize Approach Augmented with Wikipedia Knowledge for Biomedical Lay Summarization

Zhiwen You[1], Shruthan Radhakrishna[2], Shufan Ming[1], Halil Kilicoglu[1]
[1]School of Information Sciences, University of Illinois Urbana-Champaign, USA
[2]Department of Computer Science, University of Illinois Urbana-Champaign, USA
{zhiweny2, sr73, shufanm2, halil}@illinois.edu

Abstract

As the number of scientific publications is growing at a rapid pace, it is difficult for laypeople to keep track of and understand the latest scientific advances, especially in the biomedical domain. While the summarization of scientific publications has been widely studied, research on summarization targeting laypeople has remained scarce. In this study, considering the lengthy input of biomedical articles, we have developed a lay summarization system through an extract-then-summarize framework with large language models (LLMs) to summarize biomedical articles for laypeople. Using a fine-tuned GPT-3.5 model, our approach achieves the highest overall ranking and demonstrates the best relevance performance in the BioLaySumm 2024 shared task[1].

1 Introduction

New research in the biomedical field is often reported in the latest scientific articles and plays a crucial role in improving human health and wellbeing. However, the complex terminology and scientific language used in these publications can be challenging to understand for those without extensive knowledge of the field. The Biomedical Lay Summarization task (BioLaySumm) (Goldsack et al., 2024) addresses this challenge by creating summaries that are easier to read and understand, specifically tailored for readers unfamiliar with biomedical studies. Unlike traditional text summarization, which aims to condense documents into brief summaries, BioLaySumm also focuses on using easy-to-understand language. This approach ensures that the summaries are less technical and more accessible, while traditional summarization tasks prioritize capturing the precise scientific terminology used in the original documents. To solve the limited size and scope issue in current lay summarization corpora, Goldsack et al. (2022) have proposed two novel lay summarization datasets, PLOS and eLife, including biomedical journal articles alongside expert- and author-written lay summaries, which form the basis for the shared task.

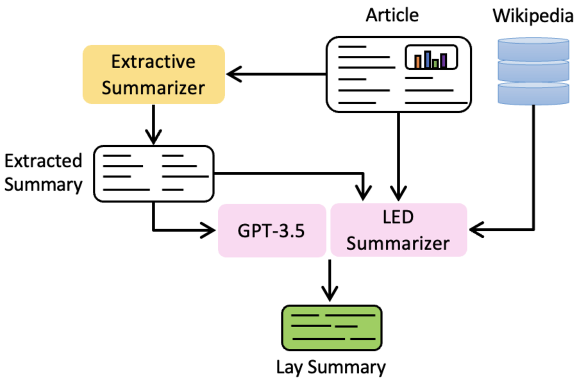


Figure 1: Our Extract-then-Summarize framework for the biomedical lay summarization task. We assess the performance of two models, GPT-3.5 and LED, in generating lay summaries. The input of the LED model includes the article sections that are ranked for relevance, Wikipedia knowledge, and an extractive summary. Meanwhile, the GPT-3.5 model is fine-tuned by the extractive summaries.

We submit lay summaries generated by two fine-tuned LLMs for this task (one for each dataset). Considering the constraints on input size and computational resources (i.e., one GPU with 32 GB memory), using the full length of scientific articles for LLM fine-tuning is not feasible. Therefore, we adopt an extract-then-summarize approach (Koh et al., 2022; Bajaj et al., 2021) (illustrated in Figure 1), which allows us to reduce the input length while maintaining competitive performance. We fine-tune the Longformer Encoder-Decoder (LED) model (Beltagy et al., 2020) and GPT-3.5 (OpenAI, 2024) and explore the effectiveness of combining unsupervised extractive summarization methods with a retrieval-augmented gener-

[1]Our code is available at https://github.com/zhiwenyou103/UIUC_BioNLP_BioLaySumm2024.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 132–143
August 16, 2024. ©2024 Association for Computational Linguistics

ation (RAG) approach (Guo et al., 2024) in generating lay summaries. Our experimental results on GPT-3.5 achieve the best overall ranking and the highest relevance score in the shared task.

2 Methods

We introduce our methodology for the biomedical lay summarization task (illustrated in Figure 1), including dataset description, section re-ranking, extractive summarization, RAG, GPT-3.5 fine-tuning, and evaluation measures. The detailed experimental settings are reported in Appendix A.

2.1 Datasets

We use eLife and PLOS datasets provided by Goldsack et al. (2022) for experiments. We report the average tokens of each dataset given the whole document and lay summarization in Appendix B. The lay summaries of eLife are crafted by expert editors, offering extensive abstraction and enhanced readability. Conversely, PLOS presents lay summaries written directly by the authors of articles. In terms of articles, eLife comprises peer-reviewed publications encompassing a broad spectrum of life sciences and medicine. PLOS covers journals spanning Biology, Computational Biology, Genetics, Pathogens, and Neglected Tropical Diseases (Goldsack et al., 2022).

2.2 Preprocessing

We employ two methods to preprocess the input article, aiming to reduce its length and extract salient sentences: section reordering and unsupervised extractive summarization.

2.2.1 Section Reordering

To better understand the experimental datasets, we conduct preliminary experiments to analyze which sections are most relevant to the gold standard lay summaries. We first group the headings of each article in eLife and PLOS datasets into five categories using structured section labels provided by the National Library of Medicine (NLM)[2] (See Appendix C for more details). Then, based on the results of section-level similarity comparison, we reorder the whole article in the order of abstract, background, conclusion, results, and methods sections. The results of the reordering method in the eLife validation set in Appendix C show the effectiveness of the restructured article compared with the default section order.

2.2.2 Unsupervised Extractive Summarization

Given the input length constraints and limited computing resources, fully incorporating scientific articles for model fine-tuning is impractical. To capture the essential global information of the articles, we implement two unsupervised extractive summarization approaches: a graph-based ranking method (TextRank) (Mihalcea and Tarau, 2004) and a BERT-based clustering method (Miller, 2019), to extract salient sentences from the documents.

TextRank (Mihalcea and Tarau, 2004) operates by treating text as a graph, where nodes are constructed based on lemmas, parts-of-speech tags of tokens in the text, and edges based on co-occurrence within a window. By iteratively applying a ranking algorithm similar to Google's PageRank (Brin and Page, 1998), it identifies the essential tokens, helping generate summaries or extract key information from text documents.

We use a BERT-based clustering approach (Miller, 2019) for unsupervised extractive summarization. It starts by dividing the article into segments using the LangChain[3] `NLTKTextSplitter` API. Next, we apply a pre-trained embedding model PubMedBERT[4] $\mathcal{E}$ deployed through SentenceTransformers[5] to encode sentences from both lay summaries and segmented passages. We calculate the cosine similarity between these embeddings to create a contrastive learning dataset $\mathcal{C}$, essential for fine-tuning the embedding model adapted for lay summarization tasks. Specifically, pairs with cosine similarity scores above 0.9 are considered positive, indicating high relevance, while those with scores below 0.01 are negative, indicating minimal relevance. We then fine-tune $\mathcal{E}$ with $\mathcal{C}$ through a contrastive loss. Appendix B presents the created dataset statistics of $\mathcal{C}$. Following the method proposed by Miller (2019), we apply a K-means clustering approach to group sentences with the same themes and find the sentences closest to the cluster's centroids as salient sentences. We extract the top 50 closest sentences from all clusters, each with a maximum length of 256

[2] https://lhncbc.nlm.nih.gov/ii/areas/structured-abstracts/downloads/Structured-Abstracts-Labels-102615.txt

[3] https://www.langchain.com/

[4] https://huggingface.co/NeuML/pubmedbert-base-embeddings

[5] https://huggingface.co/sentence-transformers

Models	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
BART (baseline)	0.4696	0.1395	0.4358	0.8623	12.0359	10.1475	13.4852	48.0963	0.7788	0.7026
GPT-3.5	0.4855	**0.1569**	0.455	**0.8677**	**11.7535**	**9.3388**	**13.3642**	52.8504	**0.8004**	**0.7338**
PubMed$_{LED}$	**0.4926**	0.1563	**0.4576**	0.8585	12.4500	9.8969	13.4096	**63.7736**	0.7576	0.6828

Table 1: Performance of our final submission and the baseline models on the test sets of both eLife and PLOS datasets. BART represents the baseline model proposed by the BioLaySumm organisers. GPT-3.5 is our final submission on the leaderboard, and PubMed$_{\text{LED}}$ is an open-source model for comparison. PubMed$_{\text{LED}}$ model tuning involves reordered sections, extractive summary, and RAG (i.e., DPR and Wikipedia definition retrieval).

tokens, as an extractive summary.

These two extractive summarization approaches reduce the overall document length, capture the essential global context, and facilitate efficient model fine-tuning.

2.3 Retrieval-Augmented Generation

To simplify model-generated lay summaries, we resort to external knowledge due to the limited background information in the datasets. First, we use a keyword-based definition retrieval method to extract definitions from the Wikipedia dataset (Guo et al., 2024) through string matching. Specifically, we employ KeyBERT[6] to extract the top 10 keywords from each article's abstract using BERT embeddings. Then, we use dataset-provided and extracted keywords to retrieve short definitions from the Wikidata-based dataset (Guo et al., 2024). We use the Wikipedia API via LangChain for extended definitions if no results are found. We concatenate the retrieved information with the input article.

Additionally, we apply an embedding-based method to extract relevant information by selecting passages from the "wiki_dpr" dataset, which contains 21 million 100-word passages from Wikipedia (Lewis et al., 2020b). Using the pre-trained dense retrieval (DPR) component of the RAG model, we retrieve the five most relevant passages to integrate into our generation tasks.

2.4 GPT-3.5 Fine-Tuning

We also experiment with GPT-3.5-turbo[7], a large-scale closed-source model from OpenAI. Our experiments, along with findings from Turbitt et al. (2023), demonstrate performance below the baseline in zero-shot and few-shot prompting settings. Consequently, we investigate fine-tuning the model using the OpenAI API[8]. To minimize API costs, we employ the extract-then-summarize approach, utilizing TextRank to extract key sentences from the full text, which are then fed into the GPT model for summary generation. Our results indicate that fine-tuning on small random samples (100 to 400 examples) is adequate to achieve high performance for the task. For our final submission, we extract 40 sentences per article using TextRank and fine-tune separate models for each dataset using random samples of 200 articles.

2.5 Evaluation

We assess the performance of our model using the official evaluation scripts provided by the organizers (Goldsack et al., 2023), employing various automatic metrics related to relevance, readability, and factuality. Relevance is measured through ROUGE (1, 2, and L) (Lin, 2004) and BERTScore (Zhang et al.). Readability metrics include the Flesch-Kincaid Grade Level (FKGL) (Kincaid et al., 1975), Dale-Chall Readability Score (DCRS) (Dale and Chall, 1948), Coleman-Liau Index (CLI) (Coleman and Liau, 1975), and LENS (Maddela et al., 2023). Notably, lower FKGL, DCRS, and CLI scores signify improved readability. Factuality evaluation incorporates AlignScore (Zha et al., 2023) and SummaC (Laban et al., 2022).

3 Results and Analysis

We present the evaluation results of our methods in the leaderboard in Table 1. Fine-tuning GPT-3.5 ranks best overall among our submissions and outperforms the baseline BART model in all aspects. Additionally, we experiment with different numbers of sentences being extracted by the TextRank (Table 2) and different numbers of training examples on the eLife validation set (Table 3). We observe no significant improvement over various evaluation metrics when increasing the number of TextRank sentences beyond 40 and the training set size beyond 200 examples.

Despite GPT-3.5's better performance over the smaller encoder-decoder models in most evaluation

[6] https://github.com/MaartenGr/KeyBERT

[7] https://platform.openai.com/docs/models/gpt-3-5-turbo

[8] https://openai.com/api/

# TextRank	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
20	0.4923	0.1498	0.4696	0.8647	9.1871	**7.4619**	10.6136	60.7699	0.6408	0.5460
30	0.5024	0.1525	0.4797	0.8647	9.1804	7.5366	10.5684	59.8206	0.6412	0.5352
40	0.5134	0.1566	0.4897	**0.8677**	**9.0398**	7.6474	**10.5426**	**61.9627**	0.6502	**0.5524**
50	0.5094	0.1563	0.4869	0.8661	9.1896	7.5420	10.5649	60.9104	0.6449	0.5466
100	**0.5151**	**0.1582**	**0.4907**	0.8667	9.5302	7.8078	10.8467	60.9124	**0.6563**	0.5432

Table 2: Ablation study of the number of sentences extracted by the TextRank for GPT-3.5 fine-tuning. The model is fine-tuned on 200 examples in each case, and evaluation is performed on the eLife validation set.

# Examples	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
w/o FT	0.3430	0.0786	0.3171	0.8360	15.6929	11.2094	17.4336	**65.3938**	**0.7467**	0.5106
100	0.49024	0.1449	0.4667	0.8625	9.9315	8.0081	11.2464	57.2724	0.6784	**0.5936**
200	**0.5134**	0.1566	**0.4897**	**0.8677**	**9.0398**	**7.6474**	**10.5426**	61.9627	0.6502	0.5524
400	0.5120	**0.1568**	0.4888	0.8673	9.3402	7.7173	10.7939	61.6123	0.6682	0.5580

Table 3: Ablation study of the number of training examples used to fine-tune GPT-3.5. All models use 40 sentences extracted by the TextRank. We apply the random seed as 42 for selecting examples. The w/o FT case uses a zero-shot prompting method to generate lay summaries. The prompt template for GPT-3.5 is provided in Appendix D.

Models	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
LED-base$_{4k}$	0.4724	0.1326	0.4503	0.8462	9.4983	**7.8822**	**10.0612**	67.4096	0.6282	**0.6093**
+ *DPR*	0.4822	0.1357	0.4592	0.8467	9.2784	7.9563	10.2329	68.4289	0.5760	0.5965
+ *Def*	0.4791	0.1347	0.4577	0.8466	9.2589	7.9413	10.2708	69.1213	0.5827	0.5981
+ *Ext*	0.4823	0.1347	0.4599	0.8466	**9.2203**	7.9210	10.2442	69.1202	0.6141	0.5902
+ *TR*	0.4818	0.1342	0.4601	0.8468	9.3755	7.9595	10.3095	68.5750	0.6129	0.5920
+ *All*	0.4810	0.1353	0.4582	0.8470	9.3195	7.9153	10.3041	68.6305	0.6150	0.5883
PubMed$_{LED4k}$	0.5070	0.1507	0.4770	0.8519	11.5237	8.9008	11.5916	69.7507	**0.6442**	0.5887
+ *All*$_{PubMed}$	**0.5140**	**0.1550**	**0.4868**	**0.8520**	10.3212	8.2847	10.6131	**70.7518**	0.6341	0.5883

Table 4: Ablation study of different model components on the eLife validation set. We use the same reordered sections of each article as base input. We apply PubMed LED large model for PubMed$_{LED4k}$ and All$_{PubMed}$ settings, and use LED-base model for all the other experiments due to limited computing resources. We report the result of PubMed$_{\text{LED}}$ model using *All* setting in Table 1.

aspects (Table 1), open-source models have some advantages, including reduced costs, the ability to fine-tune on various datasets, and reproducibility. Therefore, we conduct ablation studies on fine-tuning open-source models in Table 4. Initially, we compare two baseline configurations: the LED base model (LED-base$_{4k}$) and the PubMed LED large model (PubMed$_{LED4k}$), both using the top three sections as base input. We then apply the functional modules described in Section 2 to these baseline settings to assess the effectiveness of each component. In Table 4, *DPR* and *Def* refer to the RAG methods outlined in Section 2.3, which involve dense retrieval and entity-based definition retrieval from Wikipedia, respectively. *Ext* and *TR* denote the use of BERT-based unsupervised extractive summarization and TextRank, as introduced in Section 2.2.2. The term "*All*" represents the integration of all components, in the sequence of the top three sections, *Ext*, *DPR*, and *Def*, as input for fine-tuning. The results indicate the PubMed LED large model achieves the highest LENS and relevance scores when all components are included. However, the readability scores do not surpass those of the LED base model.

Meanwhile, we notice that an article's abstract achieves the highest factuality scores compared to the gold lay summary. Therefore, we explore the possibility of aligning the lay summaries generated by the PubMed LED model more closely with the article's abstracts through GPT-4 post-processing. However, our experimental results indicate no apparent improvement across most evaluation metrics when using GPT-4 to enhance the alignment of the generated lay summaries with the abstracts (experimental details in Appendix E).

4 Discussion and Conclusion

Applying the extract-then-summarize framework to fine-tune GPT-3.5 demonstrates superior performance in the biomedical lay summarization task compared to LED-based fine-tuning. The ablation study indicates that incorporating external knowledge during model fine-tuning slightly enhances relevance metrics in the experiments of the LED-base model. However, it negatively impacts factuality scores, similar to the results observed when using extractive summarization and PubMed LED large model. None of the components enhance the factuality scores compared to the baseline settings, although there are improvements in relevance and readability scores (Table 4). We hypothesize that the external knowledge generated by RAG methods might contain noisy data, potentially affecting the factuality metrics. Additionally, the extractive summarizer may produce sentences with less contextual coherence than the original article, hindering the model's ability to understand causal information during fine-tuning. While increasing the model size enhances relevance scores, it decreases readability and factuality from the LED base model to the PubMed LED large model.

The case study detailed in Appendix F reveals that the GPT-3.5 and PubMed LED models produce unrelated information when creating lay summaries compared to the gold lay summary. Notably, GPT-3.5 produces longer summaries than the PubMed LED model despite both models having the same maximum decoding token limit of 512. Consequently, while GPT-3.5 includes more relevant sentences that closely match the original summary, it also introduces more irrelevant content.

Overall, fine-tuning GPT-3.5 with extractive summaries achieves the best overall ranking and highest relevance score in the BioLaySumm 2024 shared task, demonstrating the effectiveness of using key sentences from the article for LLM fine-tuning. The PubMed LED model, with additional functional components, also shows competitive results compared to GPT-3.5. Meanwhile, our findings using the PubMed LED model suggest a promising direction for future studies to develop advanced modules that combine extractive summarization and RAG to generate lay summaries, especially in improving the relevance scores and enhancing the accessibility of biomedical research.

Limitations

Our study's limitations are as follows: 1) We conduct experiments using LED-based models for only one epoch with a small batch size due to time and computational constraints. We hypothesize that the model's performance could vary with more tuning epochs. 2) Our section reordering method may miss sections that do not match the NLM dictionary of section names, potentially impacting the model's performance by omitting important content from articles. The proportions of mismatched sections are detailed in Appendix C. 3) The unsupervised extractive summarization methods used in this study are not tailored for lay summarization tasks, which may result in less relevant extraction. We suggest that developing a task-specific extractor could be a promising direction for future work. 4) We apply only two RAG methods in our experiments and concatenate the retrieved knowledge at the end of the input. The quality of the retrieved information was not filtered or verified, which may negatively impact fine-tuning performance. 5) Our method uses GPT-3.5 from OpenAI, which may not be fully reproducible since GPT-3.5 is a closed-source model and may update without unambiguous versions.

Acknowledgement

This work is partially supported by the National Library of Medicine (NLM) of the National Institutes of Health under the award number R01LM014292.

References

Ahsaas Bajaj, Pavitra Dangati, Kalpesh Krishna, Pradhiksha Ashok Kumar, Rheeya Uppaal, Bradford Windsor, Eliot Brenner, Dominic Dotterrer, Rajarshi Das, and Andrew Mccallum. 2021. Long document summarization in a low resource setting using pretrained language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop*, pages 71–80.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Sergey Brin and Lawrence Page. 1998. The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 30:107–117.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*.

Meri Coleman and Ta Lin Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283.

Edgar Dale and Jeanne S Chall. 1948. A formula for predicting readability: Instructions. *Educational research bulletin*, pages 37–54.

Tomas Goldsack, Zheheng Luo, Qianqian Xie, Carolina Scarton, Matthew Shardlow, Sophia Ananiadou, and Chenghua Lin. 2023. Overview of the biolaysumm 2023 shared task on lay summarization of biomedical research articles. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 468–477.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yue Guo, Wei Qiu, Gondy Leroy, Sheng Wang, and Trevor Cohen. 2024. Retrieval augmentation of large language models for lay language generation. *Journal of Biomedical Informatics*, 149:104580.

J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Huan Yee Koh, Jiaxin Ju, He Zhang, Ming Liu, and Shirui Pan. 2022. How far are we from robust long abstractive summarization? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2682–2698.

Philippe Laban, Tobias Schnabel, Paul N Bennett, and Marti A Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020b. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. LENS: A Learnable Evaluation Metric for Text Simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16383–16408.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Derek Miller. 2019. Leveraging BERT for extractive text summarization on lectures. *arXiv preprint arXiv:1906.04165*.

OpenAI. 2024. gpt-3.5-turbo-1106. https://platform.openai.com/docs/models/gpt-3-5-turbo. Accessed: 2024-05-12.

Oisín Turbitt, Robert Bevan, and Mouhamad Aboshokor. 2023. MDC at BioLaySumm task 1: Evaluating GPT models for biomedical lay summarization. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 611–619, Toronto, Canada. Association for Computational Linguistics.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating Factual Consistency with A Unified Alignment Function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Experimental Settings

We apply two baseline model types in our experiments: LED base[9] (`allenai/led-base-16384`) and PubMed LED large[10] (`patrickvonplaten/led-large-16384-pubmed`) models. Our final submission uses GPT-3.5 as the base model[11].

Longformer Encoder-Decoder (LED). LED model is initialized from BART-base (Lewis et al.,

[9] https://huggingface.co/allenai/led-base-16384

[10] https://huggingface.co/patrickvonplaten/led-large-16384-pubmed

[11] https://platform.openai.com/docs/models/gpt-3-5-turbo

Models	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
LED$_{16k}$	0.4838	0.1378	0.4598	0.8475	9.5573	8.0395	10.2088	69.0407	0.6176	0.5837
LED$_{8k}$	0.4746	0.1342	0.4522	0.8463	9.5950	7.9085	10.2432	67.5068	0.6315	0.5903
LED$_{8k}$*	0.4750	0.1348	0.4530	0.8471	9.2274	7.9494	10.2126	69.8675	0.6391	0.6036
LED$_{4k}^{\dagger}$	0.4724	0.1326	0.4503	0.8462	9.4983	7.8822	10.0612	67.4096	0.6282	0.6093

Table 5: Performance comparison of various input lengths in the eLife dataset. All experiments are conducted under `led-base-16384`. 16k, 8k, and 4k are the maximum length of the model's input. * indicates that we restructure the input document in the order of abstract-background-conclusion-results-methods. † denotes we only input an article's abstract, background, and conclusion in model tuning.

Dataset	Section	Avg. Length (Train)	Avg. Length (Val)	Avg. Length (Test)
eLife	Article	13,942	13,705	11,683
	Lay Summary	437	445	-
PLOS	Article	8,963	8,925	9,039
	Lay Summary	239	239	-

Table 6: The average length of eLife and PLOS calculated by an average number of tokens. Article represents the full document of each article. Lay Summary is the gold summary of each article.

2020a) as both models share the same architecture, with a maximum input length of 16,384 tokens.

PubMed LED large. PubMed LED large model is fine-tuned on the PubMed Summarization dataset (Cohan et al., 2018) through the checkpoint of `led-large-16384`.

GPT-3.5. `GPT-3.5-turbo-1106` is fine-tuned on small random samples from the training datasets using the API provided by OpenAI.

We fine-tune the LED base and PubMed LED large models for 1 epoch and set batch size as 4. The maximum length of the decoder is 512. All experiments are conducted through one NVIDIA Tesla V100-32GB GPU. Considering memory efficiency, we use the default learning rate as 5e-5 for Adam optimization and set the floating point to 16 (i.e., fp16=True). For GPT-3.5 model fine-tuning[12], we apply the default API hyper-parameters, along with default values (i.e., epochs=3, batch size=1, learning rate multiplier=8).

We compare different input text lengths using the LED-base model in Table 5. The results indicate that decreasing the input length affects the relevance scores. Specifically, LED$_{16k}$, LED$_{8k}$, and LED$_{4k}^{\dagger}$ show a consistent decrease in relevance scores, while other evaluation metrics exhibit fluctuations. Notably, LED$_{16k}$ achieves the lowest factuality scores compared to other settings, suggesting a need to reduce the length of the input article to help the model capture more factual information. The encoder maximum length we use for both eLife and PLOS datasets in Table 1 is 8,192 tokens due to the computing limitation.

Dataset	# Train	# Validation	# Test
eLife	4,346	241	142
PLOS	24,773	1,376	142

Table 7: Statistics of LaySumm datasets.

Dataset	Split	# Positive Pairs	# Negative Pairs
eLife	Train	16,210	16,210
	Val	912	912
PLOS	Train	17,910	17,910
	Val	1,070	1,070

Table 8: Statistics of contrastive learning datasets. To balance the created datasets, we sample the same number of positive and negative pairs for each dataset.

Dataset	Train %	Val %
eLife	1.6 %	1.2 %
PLOS	0.6 %	0.4 %

Table 9: Unmatched section headings for eLife and PLOS datasets in section selection.

B Dataset Statistics

We report the average length of the article and lay summary in Table 6, as well as the statistics of two datasets in Table 7. As reported by Goldsack et al. (2022), there are no gold lay summaries for the test sets of both datasets for fair competition. In Table 8,

[12] https://platform.openai.com/docs/guides/fine-tuning

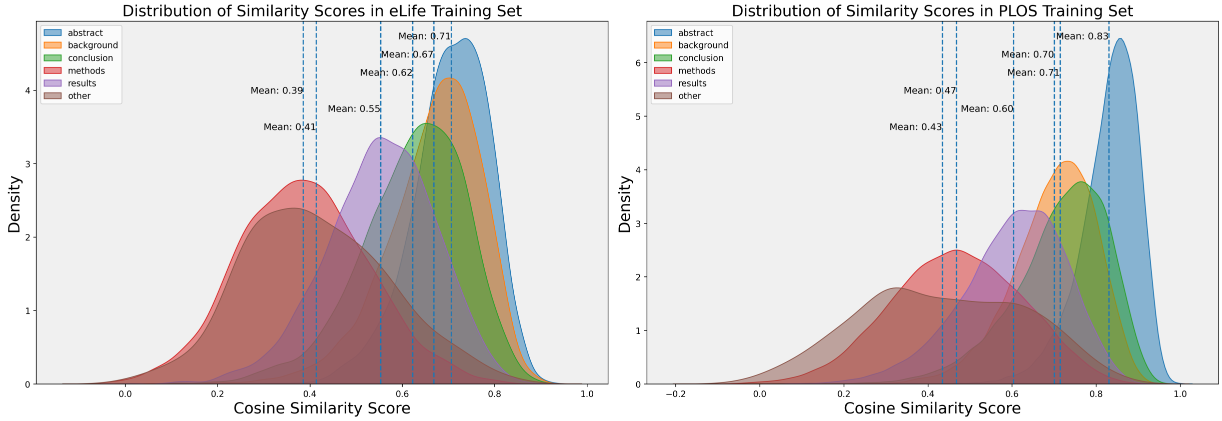


Figure 2: Comparison of section relevance in eLife and PLOS training sets grouped by NLM structured section labels. Density refers to the estimated probability density function of the cosine similarity scores for each section heading with the gold lay summary.

Models	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
LED$_{original}$	**0.4812**	**0.1355**	**0.4586**	0.8466	9.2523	**7.9069**	10.2308	67.9287	0.6224	0.5994
LED$_{ordered}$	0.4750	0.1348	0.4530	**0.8471**	**9.2274**	7.9494	**10.2126**	**69.8675**	**0.6391**	**0.6036**

Table 10: Evaluation of reordering sections in model tuning in the validation set of eLife. We set the input length of both models as 8192 tokens for equal comparison.

we show the contrastive learning datasets statistics for fine-tuning the embedding model introduced in Section 2.2.2.

C Section Relevance

As introduced in Section 2.2.1, we apply a section re-ranking strategy in our experiments to deal with long input lengths. We first pair the headings that appear in each dataset with the structured abstract section labels provided by NLM, which contains 3,032 section labels and 5 corresponding broader NLM categories: BACKGROUND, OBJECTIVE, METHODS, RESULTS, and CONCLUSIONS. We report the heading matching proportions in Table 9. Specifically, for the eLife dataset, we identify 339 unmatched headings in the training set and 14 in the validation set, out of 21,315 and 1,158 headings, respectively. The PLOS dataset has 833 unmatched headings in the training set and 28 in the validation set, with overall totals of 122,873 and 6,800 headings, respectively. Notably, no OBJECTIVE sections are matched in either the eLife or PLOS datasets. In these cases, we concatenate the unmatched sections to the end of the article. Subsequently, we rank the sections by calculating the cosine similarity between each section's content and the lay summary. We employ a pre-trained sentence transformer embedding model, `all-MiniLM-L6-v2`[13], to encode both the lay summary and each section, allowing us to compute similarity scores. Figure 2 depicts the section relevance distribution in eLife and PLOS training sets. Our findings indicate that the most relevant sections for both the eLife and PLOS datasets are the "Abstract", "Background", and "Conclusion", while the "Method" and mismatched "Other" sections are found to be less relevant.

Additionally, in Table 10, we compare the effectiveness of section reordering in an article by assessing the performance of models using ordered sections versus the original order. We use the LED-8k model on the eLife validation set for this evaluation. Specifically, we directly truncate the input as the natural order of the original dataset for the LED$_{original}$ model. Given our exploration of sections' relevance with gold lay summary (Figure 2), we re-rank the sections based on the order of abstract-background-conclusion-results-methods. Therefore, we reorder the input of the LED$_{re\text{-}order}$ model given the above order and truncate the model with an input limit of 8,192 tokens. Table 10 demonstrates that most evaluation scores improve with ordered sections as input, whereas ROUGE scores and DCRS show a decline.

[13] `https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2`

Models	R-1	R-2	R-L	BERTScore	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
Abs.	0.3189	0.0701	0.2934	0.8390	15.5000	11.7386	17.5873	38.3429	**0.9935**	**0.9488**
LaySum	**1.0000**	**1.0000**	**1.0000**	**1.0000**	10.8295	8.9364	12.4921	61.5753	0.5959	0.4943
P-LED$_L$	0.5140	0.1550	0.4868	0.8520	10.3212	8.2847	10.6131	70.7518	0.6341	0.5883
GPT-4	0.4979	0.1316	0.4668	0.8521	13.1813	9.9235	14.3912	70.3339	0.6669	0.5170

Table 11: Post-processing results in eLife validation set. Abs. and LaySum use the article's abstract and gold lay summary for evaluation. P-LED denotes PubMed LED large model (i.e., `patrickvonplaten/led-large-16384-pubmed`) and sets the input length as 8,192 tokens. GPT-4 represents the results using GPT-4 for post-processing.

D Prompt Template of GPT-3.5

We provide the prompt template for fine-tuning GPT-3.5 at the end of this section. The `### Article` represents the output of extractive summarization using TextRank introduced in Section 2.2.2. The `### Summary` denotes the gold lay summary of each article.

In the setting without fine-tuning (w/o FT), as introduced in Table 3, we use extractive summaries to prompt GPT-3.5 using the same template as fine-tuning GPT-3.5. The key difference in the w/o FT approach is that it does not include the gold-standard lay summary.

```
System:
Generate a lay summary of this biomedical
article

User:
### Article:
Cell-fate reprogramming is at the heart of
development...
### Summary:
```

E Post-Processing using GPT-4

We observe the abstracts of articles achieve higher factuality scores compared to the gold lay summaries. As illustrated in Table 11, abstracts attain the highest factuality scores, while gold lay summaries achieve full relevance scores. We assess whether an LLM can reassemble model-generated lay summaries to resemble the articles' abstracts more closely, thereby improving the factuality of the lay summaries. The input of GPT-4 includes a prompt template, original abstract, and PubMed LED model generated lay summary. While the AlignScore improves when using GPT-4 compared to the original settings, most other evaluation metrics, particularly readability scores, show a decline.

Due to the lack of factually accurate lay summaries as references to prompt or fine-tune GPT-4, the experiment is conducted solely under a zero-shot setting. We conclude that using zero-shot prompting with GPT-4 does not enhance the factuality of the generated lay summaries.

The prompt template used for GPT-4 is as follows:

```
System:
You will be provided with a biomedical
abstract and a corresponding lay summary.

Your task is to enhance the lay summary by integrating factual information
from the abstract. Consider the abstract as
an additional reference.

Make sure to keep the same wording as the provided lay summary but add
more factual information. Do not change
any factual information. Do not reduce the
readability and relevance of your enhanced
lay summary. Do not make up information.

Keep your enhanced lay summary
roughly the same length as the provided
lay summary.

User:
Abstract: Cell-fate reprograming is at the
heart of development...
Lay Summary: Genes are the building
blocks of life...
```

F Case Study

To provide a more straightforward comparison of model-generated lay summaries, we conduct a case study comparing the generated lay summaries of GPT-3.5 and the PubMed LED large models in the test sets of eLife and PLOS in Table 12 and Table 13. We randomly select an article with ID elife-78005-v1 from eLife and journal.pgen.1008255 from PLOS. Our findings reveal that both the fine-tuned GPT-3.5 and the PubMed LED model generate irrelevant information not mentioned in the gold lay summary in both datasets. However, GPT-3.5 produces more informative sentences than the PubMed LED large model.

Gold	When light enters the eyes , it is focused onto the retina , a thin layer of brain tissue at the back of the eye . The retina converts light information into electrical signals that are transmitted to the rest of the brain to perceive vision . Unlike the rest of the brain , this light-processing tissue can continue working even when removed from an animal , making it easier for scientists to study how the retina works . This has helped it become one of the best-understood parts of the brain . Most knowledge of retinal signal processing comes from studies of isolated retinas . However , it was still unclear if these samples behave the same way as they do in live animals , and whether findings in isolated retinas apply to natural visual processing in an awake state . To determine this , Boissonnet et al . compared the visual responses of the retina in awake mice , anesthetised mice and when isolated from mice . Measurements of retinal electrical signals showed that awake mice responded to light substantially more quickly and strongly than the others . Computational analysis suggested that the amount of information carried to the brain was largely comparable across the different subjects , but the retina in awake mice used more energy . The findings indicate that further studies are needed to better understand how the retina processes visual information in awake animals , rather than just in isolated conditions . Progressing this understanding could ultimately help to develop prosthetic devices that can act as a retina in the future .
GPT-3.5	The retina is a thin layer of tissue at the back of the eye that contains cells that are sensitive to light . These cells convert light into electrical signals that are sent to the brain , where they are processed to create the images that we see . The retina is one of the best understood parts of the brain , and many of the principles that govern how it works have been discovered by studying the retinas of animals in the lab . However , it is not clear how much of this knowledge applies to the retina in a living animal . This is because most of the studies have been done on retinas that have been removed from the eye and kept alive in a dish , or on retinas of animals that have been anesthetized . To investigate how the retina works in a living animal , Li et al . recorded the electrical activity of the retinas of mice while the animals were awake and while they were anesthetized . The experiments showed that the retinas of awake mice were more active than those of anesthetized mice . They also responded more quickly to changes in light and were able to detect a wider range of light levels . However , the retinas of awake mice were less efficient at transmitting information to the brain . This is because the increased activity in the retina did not result in a proportional increase in the amount of information that was sent to the brain . The findings of Li et al . suggest that the retina may use a different strategy to process information in living animals than it does in the lab . This is important because many of the principles that have been discovered by studying the retina in the lab are used to develop artificial retinas for people with vision loss . The findings of Li et al . suggest that these artificial retinas may need to be designed to work differently in living animals than they do in the lab .
PubMed$_{LED}$	The retina is the part of the brain that receives information from the eye and processes it into a visual image. The retina is made up of many different types of cells, but the activity of the retinal nerve cells is the most studied. The retinas are the most sensitive to light and are responsible for detecting color, contrast and motion. The activity of these cells is then processed by the brain, which then sends the information to the retina. However, it is not clear what exactly the retina tells the brain in awake animals. To investigate this question, Vlasiuk et al. recorded the electrical activity of retinal cells in the eyes of mice while they were awake, anesthetized and under anesthesia. The experiments show that the retinas of awake mice show faster firing rates and a larger range of responses than those of anesthetics-treated mice. This suggests that the retina uses a dense code of information in the retina, rather than a sparse code as had been assumed. The results also suggest that the brain uses dense coding in vivo. The next challenge will be to understand how the retina works in the brain and how it can be used to help us to understand the brain's role in visual processing. This could help us understand how we perceive the world around us, and how the brain responds to the world we see.

Table 12: A case study of article ID elife-78005-v1 generated by GPT-3.5 and PubMed LED large model on the eLife test set. Information highlighted in green indicates details mentioned in the gold lay summary, while red indicates details not included in the gold lay summary.

Gold	Gene inactivation in mice can be used to identify genes that are involved in important biological processes and that may contribute to disease . We used this approach to study the Ric-8B gene , which is highly conserved in mammals , including humans . We found that Ric-8B is essential for embryogenesis and for the proper development of the nervous system . Ric-8B mutant mouse embryos are smaller than their wild type littermates and show neural tube defects at the cranial region . This approach also allowed us to identify the biological pathways that potentially contribute to the observed phenotypes , and uncover a novel role for Ric-8B in the mTORC2 signaling pathway . mTORC2 plays particular important roles in the adult brain , and has been implicated in neurological disorders . Our mutant mice provide a model to study the complex molecular and cellular processes underlying the interplay between Ric-8B and mTORC2 in neuronal function .
GPT-3.5	mTOR is a serine/threonine protein kinase that is involved in a series of critical cellular processes . mTOR can be found in two functionally distinct complexes , mTORC1 and mTORC2 . In contrast to mTORC1 , little is known about the mechanisms that regulate mTORC2 . Here we show that mTORC2 activity is reduced in mice with a hypomorphic mutation of the Ric-8B gene . Ric-8B is a highly conserved protein that acts as a non-canonical guanine nucleotide exchange factor (GEF) for heterotrimeric Gs/olf type subunits . We found that Ric-8B hypomorph embryos are smaller than their wild type littermates , fail to close the neural tube in the cephalic region and die during mid-embryogenesis . Comparative transcriptome analysis revealed that signaling pathways involving GPCRs and G proteins are dysregulated in the Ric-8B mutant embryos . Interestingly , this analysis also revealed an unexpected impairment of the mTOR signaling pathway . Phosphorylation of Akt at Ser473 is downregulated in the Ric-8B mutant embryos , indicating a decreased activity of mTORC2 . Knockdown of the endogenous Ric-8B gene in cultured cell lines leads to reduced phosphorylation levels of Akt (Ser473) , further supporting the involvement of Ric-8B in mTORC2 activity . Our results reveal a crucial role for Ric-8B in development and provide novel insights into the signals that regulate mTORC2 .
PubMed$_{LED}$	mTOR, a serine/threonine protein kinase that is involved in a series of critical cellular processes, can be found in two functionally distinct complexes, mTORC1 and mTOR2. In contrast to mTOR1, little is known about the mechanisms that regulate mTORc2. Here we show that the Ric-8B gene is required for proper growth and nervous system formation during mouse embryogenesis. We found that the mutant embryos fail to close the neural tube at the cephalic region and die during mid-embryogenesis. Our results reveal a crucial role for Ric- 8B in development and provide novel insights into the signals that regulate the mTOR signaling pathway.

Table 13: A case study of article ID journal.pgen.1008255 generated by GPT-3.5 and PubMed LED large model on the eLife test set. Information highlighted in green indicates details mentioned in the gold lay summary, while red indicates details not included in the gold lay summary.

End-to-End Relation Extraction of Pharmacokinetic Estimates from the Scientific Literature

Ferran Gonzalez Hernandez[1,†,*], **Victoria C. Smith**[2,8†], **Quang Nguyen**[2,†], **José Antonio Cordero**[3], **Maria Rosa Ballester**[3,4], **Màrius Duran**[3], **Albert Solé**[3], **Palang Chotsiri**[5], **Thanaporn Wattanakul**[6], **Gill Mundin**, **Watjana Lilaonitkul**[7], **Joseph F. Standing**[8,9], **Frank Kloprogge**[10]

[1]Department of Computer Science, UCL, UK [2]Institute of Health Informatics, UCL, UK
[3]Blanquerna School of Health Sciences, Ramon Llull University, Spain [4]Institut de Recerca Sant Pau Barcelona, Spain
[5]Clinical Pharmacology, Modelling and Simulation, Parexel International, Thailand
[6]Mahidol Oxford Tropical Medicine Research Unit, Thailand [7]Global Business School for Health, UCL, UK
[8]Great Ormond Street Institute for Child Health, UCL, UK [10]Institute for Global Health, UCL, UK
[9]Department of Pharmacy, Great Ormond Street Hospital for Children, UK
[†] equal contribution [*]{ferran.hernandez.17, f.kloprogge}@ucl.ac.uk

Abstract

The lack of comprehensive and standardised databases containing Pharmacokinetic (PK) parameters presents a challenge in the drug development pipeline. Efficiently managing the increasing volume of published PK Parameters requires automated approaches that centralise information from diverse studies. In this work, we present the Pharmacokinetic Relation Extraction Dataset (PRED), a novel, manually curated corpus developed by pharmacometricians and NLP specialists, covering multiple types of PK parameters and numerical expressions reported in open-access scientific articles. PRED covers annotations for various entities and relations involved in PK parameter measurements from 3,600 sentences. We also introduce an end-to-end relation extraction model based on BioBERT, which is trained with joint named entity recognition (NER) and relation extraction objectives. The optimal pipeline achieved a micro-average F1-score of 94% for NER and over 85% F1-score across all relation types. This work represents the first resource for training and evaluating models for PK end-to-end extraction across multiple parameters and study types. We make our corpus and model openly available to accelerate the construction of large PK databases and to support similar endeavours in other scientific disciplines.

1 Introduction

Pharmacokinetics (PK) aims to quantify drug exposure through the study of drug absorption, distribution, metabolism and excretion (ADME). Drug PK profiles inform the selection of drug candidates and establish therapeutically relevant doses and dosing schedules (Morgan et al., 2012; Reichel and Lienau, 2016). Population PK models, i.e. nonlinear mixed-effects models, have played a significant role over the last decades in characterising PK properties through parameterising PK time series data. This has contributed to improved accuracy of predicting PK profiles across all stages of the drug development process.

Prior data from similar drug compounds are often used to initialise Population PK models and are also relevant for pre-clinical PK predictions for novel compounds (Dearden, 2007; Berellini and Lombardo, 2019; Wang et al., 2019). However, the primary challenge in collating prior PK data is the lack of comprehensive, standardised and open-access databases of PK parameter estimates, which has been recognised as a significant limitation in the drug development pipeline (Kumar et al., 2021; Mould and Upton, 2013; Grzegorzewski et al., 2021; Wang et al., 2009). Existing databases (Grzegorzewski et al., 2021; Wong et al., 2019) are manually curated from scientific literature and are limited to a few drugs. Consequently, researchers must manually compile PK information from the scientific literature (Grzegorzewski et al., 2021; Lombardo et al., 2018). The ability to automatically extract and centralise PK data from the scientific literature is of great interest to solidify existing PK knowledge and improve parameter predictions.

Annotated biomedical datasets have facilitated the development of state-of-the-art models for identifying many biomedical entities and their relationships in free text. However, no such annotated data exists for PK. In this work, we present a new dedicated corpus for Named Entity Recognition (NER) and Relation Extraction (RE) of PK data from scientific articles. This corpus is manually annotated at the sentence level by domain experts and involves entities and relations between PK pa-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 144–154
August 16, 2024. ©2024 Association for Computational Linguistics

rameter names, estimated values, deviation values, units and comparative terms. We also develop an end-to-end relation extraction architecture based on adapting the SpERT model (Eberts and Ulges, 2019) and training it on our corpus to assess the feasibility of automated extraction of PK parameter estimates. Our contributions are as follows:

- The PRED corpus, a publicly available corpus[1] of manually annotated entities and relations between PK parameter names, the central and deviation values, their units and comparative terms. PRED consists of 1764 entity mentions and 2016 relations annotated across 3600 sentences from scientific articles.
- A novel RE pipeline[2], trained and evaluated on PRED, for tackling the extraction of PK parameter estimates from the scientific literature. We compare architectures that model NER and RE jointly against models that optimise for a single task and assess the effect of domain-specific pre-training.

2 Related Work

Automated text mining approaches have been extensively explored regarding drugs and chemicals (Krallinger et al., 2015; Lee et al., 2020; Sung et al., 2022), drug-drug interactions (Herrero-Zazo et al., 2013; Segura Bedmar et al., 2013; Kolchinsky et al., 2013, 2015), and biochemical kinetics (e.g. enzyme kinetics) (Hakenberg et al., 2004; Spasić et al., 2009; Tsay et al., 2009). However, little research has been conducted on automatically extracting PK data from text.

Wang et al. (2009) explored pattern-based approaches for a single PK parameter for one drug. However, extending this approach to other PK parameters, drugs, and study designs becomes unfeasible due to the high diversity of surface forms. Instead, approaching PK information extraction with machine learning approaches has the potential to model a higher variability of PK parameters and relations effectively. Previously, Hernandez et al. (2021) presented an automated pipeline to identify scientific publications reporting PK parameter estimates measured *in vivo*. Subsequently, Hernandez et al. (2024) released a large annotated dataset of PK parameter mentions in the scientific literature and fine-tuned BioBERT (Lee et al., 2020) to perform NER of PK parameters. However, to our knowledge, no study has yet tackled the task of end-to-end relation extraction of PK parameter estimates, which represents a crucial step to automatically construct PK databases useful for drug development.

3 Methods

3.1 Corpus construction

The PRED corpus was developed to train and evaluate end-to-end pipelines that extract PK measurements from sentences and can be found at https://zenodo.org/records/11187303. All the relations tackled in this task appeared between entities within the same sentence.

Data Source

The following pipeline was applied to create a candidate pool of sentences. A PubMed search for *"pharmacokinetics"* was initially conducted in June 2020 to retrieve articles. The pipeline from Gonzalez Hernandez et al. (2021) retrieved 114,921 relevant publications reporting PK parameters. Out of these, 10,132 articles (8.82%) were accessible in full text from the PMC OA subset[3], while only abstracts were available for the rest. Both, abstracts and full-text articles were downloaded in XML format from PubMed[4] and PMC[5] FTP sites. The PubMed Parser (Titipat and Acuna, 2015) was used to parse the XML files, and paragraphs from the introduction section were excluded. The scispaCy sentence segmentation algorithm (Neumann et al., 2019) split abstracts and paragraphs into sentences. The resulting sets were randomly sampled to produce a candidate pool of 1,443,044 sentences, with a balanced proportion of sentences from the abstract and full-text. Noticeably, 16.4% of sentences from the initial candidate pool mentioned PK parameters. Therefore, a filtering protocol was applied to promote the development of a corpus with a wide variety of PK mentions and relation instances. The PK NER model from Hernandez et al. (2024) was first applied to all the candidate pool sentences. Then, we selected sentences that at least had (1) one PK mention detected by the NER model and (2) a numerical value. From the resulting pool of sentences, 3600 instances were randomly sampled

[1] https://zenodo.org/records/11187303

[2] https://github.com/PKPDAI/PKRelations

[3] https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/

[4] https://www.nlm.nih.gov/databases/download/pubmed_medline.html

[5] https://ftp.ncbi.nlm.nih.gov/pub/pmc/

without replacement and divided into 2100, 500 and 1000 instances for the training, development and test sets, respectively.

Annotation

The annotation team comprised 12 individuals with extensive PK expertise and familiarity with the different parameters and study types in PK literature. Annotation guidelines were developed and distributed to the annotators before labelling and updated as new complex cases emerged during annotation. To ensure consistency, annotations were performed in batches of 200 sentences, following a three-step procedure: (1) initial annotation by one PK expert, (2) review by another PK annotator and (3) final check focusing on span boundary consistency by an annotator with bio-NLP experience. After the second step in each batch, comments from the first and second steps were reviewed, and feedback regarding incorrect annotation patterns was given to the annotators. Inter-annotator agreement was examined using the pair-wise F_1 score on 200 sentences, and the mean was computed across each pair of annotators. For further details on the annotation guidelines and interface, please see Appendix A.

Task Definition

End-to-end RE aims to identify named entities and extract relations between them. Given some input text X, the output of any end-to-end RE system is a list of triplets in the form of (s_i, s_j, r) where $s_i, s_j \in S$ and $r \in R$ and S denote all the possible spans in X and R the set of pre-defined relation types (Zhong and Chen, 2020). Hence, the annotated data was represented as a list of sentences, each with their corresponding list of relation triplets and compared to model predictions in the same format. Because end-to-end RE systems need to (1) identify candidate spans and (2) predict relation classes for pairs of spans, this task is often decomposed into two sub-tasks:

1. Named Entity Recognition: which attempts to detect the list of entity mentions (i.e. spans) and their type $\mathcal{E} = \{PK, Units, Value, Range, Compare\}$ from the input text X.
2. Relation Extraction: which compares all pairs of spans in X and outputs a relation class for each pair $R = \{Central_{val}, Deviation_{val}, Related\}$.

For step 1, the following entities were considered and annotated at the sentence level:

1. **PK**: Mentions of parameters. This entity refers to spans mentioning PK parameters, and it is the same concept as the entity described by Hernandez et al. (2024).
2. **Units**: Spans of text corresponding to units of numerical PK estimations.
3. **Value**: Spans encapsulating numerical estimations related to PK parameters (i.e. central and deviation values).
4. **Range**: Two values defining the boundaries of a PK estimation.
5. **Compare**: Textual mentions that provided information about whether a specific value/range mention was the extreme of an estimated parameter. This entity appeared with low frequency, but it was important for detecting extracted measurements that were not central estimations of a certain parameter.

For step 2, three relations classes were considered between entities to extract structured information from raw sentences in a usable format. Please note the directionality of relations is not considered in this work as it is not necessary for the desired tabular output (see Figure 1):

1. **Central**$_{val}$[6]: This relation type happened between PK parameter mentions and their estimated values or ranges. This involved central measurements of the parameter but not measures of deviation or % of increase concerning other experimental conditions. The entities between which this relation could happen were:
   - PK ↔ Value/Range
2. **Deviation**$_{val}$[7]: This relation type informed whether a specific measurement was the deviation of a central measurement and only happened between the entities:
   - Value/Range ↔ Value/Range (involved in a $Central_{val}$ relation)
3. **Related**: This relation type complemented values/ranges with their units or compare terms and only happened between the following entities:

[6] Abbreviated as C_VAL in the annotation interface.

[7] Abbreviated as D_VAL in the annotation interface.

- Compare ↔ Value/Range
- Units ↔ Value/Range

3.2 Pipeline

Recent work has shown that sharing token representations and modelling NER and RE tasks simultaneously in a multi-task setting can enhance performance in both tasks (Bekoulis et al., 2018; Luan et al., 2019; Eberts and Ulges, 2019). This might be especially relevant in our corpus, where spans were only considered entities if they were part of a relation. For this reason, we propose an architecture to model NER and RE jointly to share encoded knowledge from both tasks.

Multi-task Architecture

Our multi-tasking architecture (illustrated in Figure 2), was inspired by the architecture in the SpBERT model developed by Eberts and Ulges (2019). The main modification was using sequential BIO labelling (Palen-Michel et al., 2021; Gu et al., 2021) instead of a span-based approach, as the PRED data does not contain overlapping spans. There was also no need to predict the directionality of relations for our work, so entity pairs were arranged in order of appearance in the original text. Finally, due to only one relation type existing between entity pairs in PRED, a softmax activation was used instead of a sigmoid activation.

Using the BERT tokenizer, an input sentence is initially tokenised into a sequence of sub-words. Then, tokens are passed through an encoder that aims to incorporate contextual information in each token's representation. The output embeddings from the encoder $(T_1, T_2, ...T_N)$ are then used to (1) recognise entities through the token classifier using the BIO scheme, (2) generate candidate pairs of predicted entities and (3) classify all pairs of recognised entities with a relation classifier. NER and RE use the same encoder to generate contextual representations of input tokens and have one task-specific classification layer for each sub-task. We assessed the effect of domain-specific pretraining by comparing BERT$_{BASE}$ (Devlin et al., 2018) and BioBERT v1.1 (Lee et al., 2020) as the encoder.

Named Entity Recognition Task

NER was treated as a sequential labelling problem where each output token representation from the encoder (T_i) was classified into one unique BIO scheme class using a feed-forward layer with a sigmoid activation function. The model was trained with cross-entropy loss over token-level labels $\mathcal{L}_{NER}$.

Relation Extraction Task

After NER is performed in a specific sentence, all potential pairs of predicted spans are arranged and filtered before going to the relation classifier. Then, each candidate entity pair was classified into one relation class $[Central_{val}, Deviation_{val}, Relation, No\ Relation]$. Following Taillé et al. (2020), the representation of those spans composed of multiple tokens was generated by max-pooling their contextual token embeddings. Given the effective results of the max-pooling strategy presented by Eberts and Ulges (2019), no other fusion functions were analysed. The input to the relation classifier $x(s_1, s_2)$ was the concatenation of the two-span representations e(s$_1$) and e(s$_2$) with their context representation c(s_1, s_2):

$$x(s_1, s_2) = [e(s_1); c(s_1, s_2); e(s_2)] \tag{1}$$

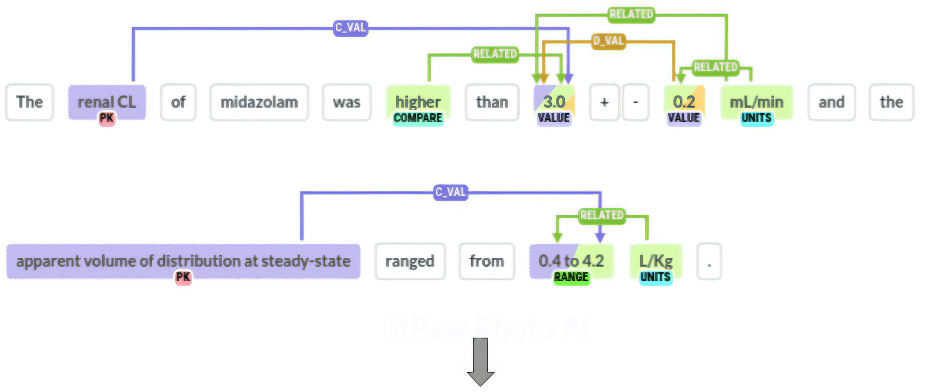
The context representation for two spans was generated by max-pooling all tokens strictly between them. If there were no tokens present between two spans $c(s_1, s_2) = 0$. Relations between entities were symmetric (non-directional) in the PRED corpus, and no overlapping spans were annotated. As a consequence, $e(s_1)$ and $e(s_2)$ were arranged according to their relative position in the sentence from left to right. Analogous to the token classifier, a single-feed forward layer was used to classify each candidate span pair. Since only one relation class could be associated between two entities, a softmax operation was used as an activation function. The model was trained with cross-entropy loss over relation classes, $\mathcal{L}_{RE}$.

Training and Optimisation

All the parameters from the encoder, the token and the relation classifier were fine-tuned during the training phase. Given sentences with annotated entities and relations, the loss was computed jointly by adding the NER and RE losses:

$$\mathcal{L} = \mathcal{L}_{NER} + \mathcal{L}_{RE} \tag{2}$$

Both losses were averaged over each batch's samples. Each batch consisted of B sentences from which samples were drawn for both classifiers. For the token classifier, the loss was computed for all



Parameter	Central measure	Units	Deviation	Deviation Units	Compare
renal CL	3.0	mL/min	0.2	mL/min	higher
apparent volume of distribution at steady-state	0.4 to 4.2	L/Kg			

Figure 1: The top panel shows a sentence where all entities and relations have been annotated. The bottom panel shows how the annotated entities and relations can be mapped into a tabular format that can be integrated into a database of PK measurements.

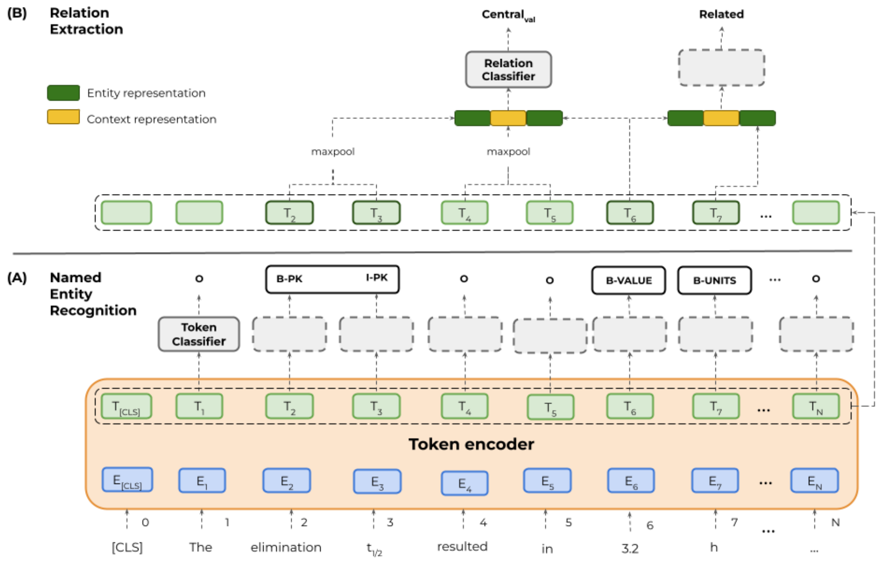


Figure 2: The model first receives a sequence of token embeddings (blue boxes, E_i) and goes through the encoder layers to generate a sequence of contextual token embeddings (green boxes, T_i), which are shared in both tasks. Then, (A) contextual token embeddings go through the token classifier (feed-forward layer) to output BIO labels that will allow recognising entities. (B) Entities and contexts (span between two entities) are represented by max-pooling their contextual token embeddings. Finally, pairs of entities are concatenated with their context representation and passed through the relation classifier (feed-forward layer).

tokens in the batch using the BIO labels. For the relation classifier, ground truth (annotated) entities were used to generate candidate pairs at training time. Negative samples ($No\ Relation$ class) were generated with all candidate entity pairs not labelled with a relation during the annotation phase. At inference time, only those entities predicted by the NER module were passed to the RE classifier instead of using ground truth entities.

Models were trained for 50 epochs and evaluated on the development set after each epoch, saving the model state with the highest $Central_{val}$ F_1 score. The maximum sequence length for all experiments was set to 256, the batch size to 8, and the learning rate to $\mu = 2e^{-5}$. The Adam Optimizer with a linear weight decay of 0.05 was used, and a dropout probability of 0.1 was applied on all layers. All experiments were run on a single GPU, NVIDIA Titan RTX (24GB).

Evaluation

Precision, Recall, and F_1 scores were computed for NER and RE. For NER, scores were based on strict matching of entity boundaries and types. F_1 scores were calculated per entity, with macro and micro-averages across entity types for overall system evaluation. In RE, focus was on F_1 score of $Central_{val}$ relations, as predicting $Deviation_{val}$ or $Related$ relations without $Central_{val}$ renders extracted data useless. Micro-averaged F_1 scores for NER and $Central_{val}$ relations for RE served as the main metrics for comparing different architectures on the PRED corpus.

4 Results and Discussion

4.1 Corpus Statistics

The main statistics for the PRED dataset are presented in Table 1. A total of 3,600 sentences were annotated, from which 56.42% contained annotated entities and relations. Sentences were evenly sampled from full-text and abstract sections. A total of 13,404 entity mentions were annotated. 12,411 relations were annotated, most coming from the $Related$ and $Central_{val}$ classes. The number of annotated $Central_{val}$ relations was over 2.5 times the number of $Deviation_{val}$, indicating that measures of deviation are not often reported along with central measures of PK parameters (only in 35.8% of cases).

4.2 Annotator Agreement

The average micro and macro-F_1 scores for NER were 88.74% and 92.36%, respectively, exhibiting high agreement on entity surfaces on the first annotation phase. For RE, the average pair-wise scores were 93.02%, 94.47% and 83.2% for $Related$, $Deviation_{val}$ and $Central_{val}$, respectively. A lower agreement was obtained between central values and their PK parameter mentions, mostly caused by disagreement on parameter span boundaries.

4.3 Multitask Model Performance

The effect of using a multi-task (MT) learning approach, jointly optimising NER and RE, was compared against a model only optimising for NER. BioBERT was used as an encoder in both cases. The MT architecture saved the model with the best $Central_{val}$ $F1$ on the development set, while micro-averaged $F1$ was used as a metric to select the best model for the no-MT experiment. Table 2 shows the NER performance on the test set for each entity type and the macro and micro-averaged $F1$ scores after ten runs of each experiment. Higher performance was obtained when using the MT architecture for all entities in the PRED corpus. Although the performance gain was relatively low ($\approx$+$\Delta F1$ 0.5%), the consistency of this gain across all entity types suggests that having the RE objective combined with NER helped the model perform better on NER. Finally, we noted higher interquartile variance for $Range$ and $Compare$ entities, which were the ones with the least number of annotations.

Although the performance gain of the MT architecture was small, such an approach also helped reduce the number of parameters required to model the task by only having one encoder. These results indicate that sharing token representations and optimising a single loss for NER and RE is beneficial for extracting PK measurements from the scientific literature compared to treating both tasks independently.

The MT solution's performance on the RE task is summarized in Table 3. Results show successful linking of deviation measurements and units in most cases. Notably, when values and units are correctly detected, their relation often requires minimal context, especially with a short distance between them. Additionally, the context between units and values typically lacked other units, simplifying extraction. Similarly, the context between central and deviation values often lacked other value entities. Therefore, with high NER performance for $Value$ and $Units$, few errors were observed for $Deviation_{val}$ and $Related$ relations. $Central_{val}$ relation showed relatively high performance, indicating consistency in dataset annotation and effective end-to-end modeling. Errors in $Central_{val}$ predictions mostly stemmed from incorrect NER predictions and sentences mentioning multiple parameters and values. However, some incorrect predictions of PK entities partially matched PK parameters, suggesting $Central_{val}$ performance could be a lower bound for PK measurement extraction. F_1 scores in Table 3 were close to or exceeded inter-annotator agreement: 93.02% vs. 93.66% for $Related$, 94.47% vs.

Table 1: Corpus statistics summarising the sentences, entities and relations in the dataset stratified by the training, development and test sets.

		Training	Development	Test	Total
Sentences	Amount #	2100	500	1000	3600
	with relations (%)	57.05	53.00	56.80	56.42[†]
	from full-text (%)	48.71	50.00	50.30	49.33[†]
Entities	PK	1890	394	856	3140
	Units	2286	474	1056	3816
	Value	3524	702	1557	5783
	Range	314	74	174	562
	Compare	51	18	34	103
Relations	$Central_{val}$	2794	571	1312	4677
	$Deviation_{val}$	1049	207	419	1675
	$Related$	3643	764	1652	6059

[†] Weighted average across datasets.

Table 2: Named Entity Recognition results on the test set for the model using multi-task (MT) learning, NER + RE, against a model only optimising for NER (no-MT). The metrics reported consider strict matching over entity mentions. Results are displayed as the median over ten runs with their interquartile variance in subscript.

	Precision		Recall		F1	
Entity	MT	no-MT	MT	no-MT	MT	no-MT
PK	$\mathbf{90.82}_{4.02}$	$89.98_{3.86}$	$\mathbf{90.57}_{3.76}$	$90.09_{3.05}$	$\mathbf{90.39}_{2.1}$	$90.02_{1.72}$
Units	$95.49_{1.87}$	$\mathbf{95.79}_{1.66}$	$\mathbf{96.17}_{2.07}$	$95.69_{3.85}$	$\mathbf{95.65}_{0.68}$	$95.56_{1.52}$
Value	$94.83_{2.78}$	$\mathbf{94.96}_{2.87}$	$\mathbf{96.18}_{3.17}$	$95.21_{5.94}$	$\mathbf{95.54}_{2.53}$	$95.04_{2.02}$
Range	$\mathbf{93.49}_{4.9}$	$93.28_{6.24}$	$\mathbf{90.26}_{8.22}$	$87.39_{10.33}$	$\mathbf{91.66}_{4.41}$	$90.4_{3.71}$
Compare	$\mathbf{88.23}_{6.81}$	$88.23_{16.99}$	$66.67_{9.09}$	$\mathbf{68.18}_{11.44}$	$\mathbf{76.53}_{5.82}$	$75.64_{8.12}$
Micro-average					$\mathbf{94.03}_{1.63}$	$93.69_{1.60}$
Macro-average					$\mathbf{90.02}_{2.23}$	$89.56_{2.45}$

93.53% for $Deviation_{val}$, 83.2% vs. 86.1% for $Central_{val}$, for inter-annotator and MT model cases, respectively. These results imply that posterior reviews and standardization of span boundaries significantly improved dataset consistency, and the model developed competes well with the expected agreement between pharmacometricians.

4.4 Encoders and Context

To analyse the effect of domain-specific pre-training in the encoder, the BioBERT model was replaced with BERT$_{BASE}$, which was pre-trained on general-domain English text. As shown in Table 4, there was a significant benefit of pre-training in biomedical text, with BioBERT exhibiting over 3% gains in all metrics compared to BERT$_{BASE}$. The largest gain ($\approx \Delta 6\%$) was observed in the $Central_{val}$ relation, indicating that pre-training on biomedical text highly improved PK NER and the understanding between parameter mentions and their measurements. These results are in line with previous findings from Wadden et al. (2019) and Eberts and Ulges (2019). Previous work on end-to-end relation extraction showed improvements between 1.1-4.4% on the SciERC and GENIA datasets with in-domain pre-training (Wadden et al., 2019; Eberts and Ulges, 2019). However, 5.9% improvement was obtained in this task for $Central_{val}$, suggesting that in-domain pre-training is particularly useful. Hence, it is likely that further pre-training on PK literature helps the model performance, and it might be a promising area for future work.

The effect of removing the local context between entities was studied. For this, the input to the RE layer was simplified to the entity embeddings. In other words, the yellow vector from Figure 2 B was removed. Table 5 shows the results of this experiment. Surprisingly, it was observed that the local context improved not only RE but also NER. Both micro and macro-F1 scores were slightly improved, suggesting that explicitly encoding local

Table 3: End-to-end relation extraction results on the test set for the MT model configuration. Results are displayed as the median over ten runs with their interquartile variance in subscript.

Relation	P	R	F_1
$Central_{val}$	$85.77_{5.04}$	$85.46_{5.07}$	$86.1_{3.49}$
$Deviation_{val}$	$92.33_{1.9}$	$94.39_{6.27}$	$93.53_{3.01}$
$Related$	$93.83_{1.69}$	$94.08_{2.51}$	$93.66_{1.52}$

Table 4: Results on the test set when using different encoder models. Results are displayed as the median over ten runs with their interquartile variance in subscript. NER metrics are the micro- and macro-averaged F_1 scores over all entities, and RE metrics are the F_1 scores for each relation class.

	NER		RE		
Encoder	macro-F_1	micro-F_1	$Related$	$Deviation_{val}$	$Central_{val}$
BERT$_{BASE}$	$85.82_{4.07}$	$90.81_{1.77}$	$89.44_{1.69}$	$90.27_{2.2}$	$80.16_{4.14}$
BioBERT	$\mathbf{90.02}_{2.23}$	$\mathbf{94.03}_{1.63}$	$\mathbf{93.66}_{1.52}$	$\mathbf{93.53}_{3.01}$	$\mathbf{86.1}_{3.49}$

Table 5: Results on the test set when using different representations as input to the relation classifier. Local context is the max-pooling of all tokens strictly between two entities. No context only used the concatenation of each entity representation in a specific relation. Results are displayed as the median over ten runs with their interquartile variance in subscript. NER metrics are the micro- and macro-averaged F_1 scores over all entities, and RE metrics are the F_1 scores for each relation class.

	NER		RE		
RE layer representaiton	macro-F_1	micro-F_1	$Related$	$Deviation_{val}$	$Central_{val}$
Local context	$\mathbf{90.02}_{2.23}$	$\mathbf{94.03}_{1.63}$	$\mathbf{93.66}_{1.52}$	$\mathbf{93.53}_{3.01}$	$\mathbf{86.1}_{3.49}$
No context (E1E2)	$89.47_{2.16}$	$93.69_{1.0}$	$91.61_{1.84}$	$90.52_{4.44}$	$81.04_{2.96}$

context between entities in RE layers can also help recognise entities better.

For relation extraction, local context seemed to provide a significant improvement for all relation types, and especially for the $Central_{val}$. This result suggests that entity embeddings might capture local information around the entity mentioned while failing to incorporate longer-range dependencies. The results obtained in this experiment are in-line with Eberts and Ulges (2019). Although recurrent and Transformer models have improved the detection of long-range dependencies in sequential inputs, the noise introduced with long context still represents a challenge in relation extraction (Eberts and Ulges, 2019; Zhong and Chen, 2020). Using this local context, the model can focus on those tokens that might be more informative about the dependencies between both entities. Nonetheless, future studies might benefit from further exploring different contextual representations for RE of PK measurements.

5 Conclusion and Future work

We introduce the PRED corpus, a large and comprehensive public corpus consisting of PK entities and relations annotated in sentences from the scientific literature. This dataset facilitates training and benchmarking models for extracting PK measurements from the scientific literature. We also train and release a new end-to-end RE model based on a BioBERT encoder. This model initially performs NER to identify spans of interest in text, followed by predicting relations between spans. Our benchmark results on the PRED dataset are promising, achieving a micro-average F1-score of 94% for NER and over 85% F1-score across all PK relation types. Our dataset and model can accelerate the construction of ADME datasets from the scientific literature, which can benefit drug development and off-label dosing.

Acknowledgements

VS and QN acknowledge support from a UCL UKRI Centre for Doctoral Training in AI-enabled Healthcare studentship (EP/S021612/1). VS acknowledges support from a studentship from the NIHR Biomedical Research Centre at University College London Hospital NHS Trust. FG would like to thank Pontus Stenetorp for helpful feedback and technical input.

References

Giannis Bekoulis, Johannes Deleu, Thomas Demeester, and Chris Develder. 2018. Joint entity recognition and relation extraction as a multi-head selection problem. *Expert Systems with Applications*, 114:34–45.

Giuliano Berellini and Franco Lombardo. 2019. An Accurate In Vitro Prediction of Human VDss Based on the Øie-Tozer Equation and Primary Physicochemical Descriptors. 3. Analysis and Assessment of Predictivity on a Large Dataset. *Drug metabolism and disposition: the biological fate of chemicals*.

John C. Dearden. 2007. In silico prediction of ADMET properties: How far have we come? *Expert Opinion on Drug Metabolism and Toxicology*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Markus Eberts and Adrian Ulges. 2019. Span-based joint entity and relation extraction with transformer pre-training. *arXiv preprint arXiv:1909.07755*.

Ferran Gonzalez Hernandez, Simon J Carter, Juha Iso-Sipilä, Paul Goldsmith, Ahmed A Almousa, Silke Gastine, Watjana Lilaonitkul, Frank Kloprogge, and Joseph F Standing. 2021. An automated approach to identify scientific publications reporting pharmacokinetic parameters. *Wellcome Open Research*, 6:88.

Jan Grzegorzewski, Janosch Brandhorst, Kathleen Green, Dimitra Eleftheriadou, Yannick Duport, Florian Barthorscht, Adrian Köller, Danny Yu Jia Ke, Sara De Angelis, and Matthias König. 2021. Pk-db: pharmacokinetics database for individualized and stratified computational modeling. *Nucleic acids research*, 49(D1):D1358–D1364.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23.

Jörg Hakenberg, Sebastian Schmeier, Axel Kowald, Edda Klipp, and Ulf Leser. 2004. Finding kinetic parameters using text mining. *OMICS A Journal of Integrative Biology*, 8(2):131–152.

Ferran Gonzalez Hernandez, Simon J Carter, Juha Iso-Sipilä, Paul Goldsmith, Ahmed A Almousa, Silke Gastine, Watjana Lilaonitkul, Frank Kloprogge, and Joseph F Standing. 2021. An automated approach to identify scientific publications reporting pharmacokinetic parameters. *Wellcome Open Research*, 6.

Ferran Gonzalez Hernandez, Quang Nguyen, Victoria C Smith, Jose Antonio Cordero, Maria Rosa Ballester, Marius Duran, Albert Sole, Palang Chotsiri, Thanaporn Wattanakul, Gill Mundin, et al. 2024. Named entity recognition of pharmacokinetic parameters in the scientific literature. *bioRxiv*, pages 2024–02.

María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. 2013. The ddi corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of biomedical informatics*, 46(5):914–920.

A. Kolchinsky, A. Lourenço, L. Li, and L. M. Rocha. 2013. Evaluation of linear classifiers on articles containing pharmacokinetic evidence of drug-drug interactions. *Pacific Symposium on Biocomputing*, pages 409–420.

Artemy Kolchinsky, Anália Lourenço, Heng-Yi Wu, Lang Li, and Luis M Rocha. 2015. Extraction of pharmacokinetic evidence of drug–drug interactions from the literature. *PloS one*, 10(5):e0122199.

Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M Lowe, et al. 2015. The chemdner corpus of chemicals and drugs and its annotation principles. *Journal of cheminformatics*, 7(1):1–17.

Vikas Kumar, Mohammad Faheem, Keun Woo Lee, et al. 2021. A decade of machine learning-based predictive models for human pharmacokinetics: Advances and challenges. *Drug discovery today*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Franco Lombardo, Giuliano Berellini, and R. Scott Obach. 2018. Trend analysis of a database of intravenous pharmacokinetic parameters in humans for 1352 drug compounds. *Drug Metabolism and Disposition*, 46(11):1466–1477.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A general framework for information extraction using dynamic span graphs. *arXiv preprint arXiv:1904.03296*.

Matthew Montani, Ines and Honnibal. 2018. Prodigy: A new annotation tool for radically efficient machine teaching. *Artificial Intelligence*, to appear.

Paul Morgan, Piet H Van Der Graaf, John Arrowsmith, Doug E Feltner, Kira S Drummond, Craig D Wegner, and Steve DA Street. 2012. Can the flow of medicines be improved? fundamental pharmacokinetic and pharmacological principles toward improving phase ii survival. *Drug discovery today*, 17(9-10):419–424.

Diane R Mould and Richard Neil Upton. 2013. Basic concepts in population modeling, simulation, and model-based drug development—part 2: introduction to pharmacokinetic modeling methods. *CPT: pharmacometrics & systems pharmacology*, 2(4):1–14.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. 2019. ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing.

Chester Palen-Michel, Nolan Holley, and Constantine Lignos. 2021. Seqscore: Addressing barriers to reproducible named entity recognition evaluation. *arXiv preprint arXiv:2107.14154*.

Andreas Reichel and Philip Lienau. 2016. Pharmacokinetics in drug discovery: an exposure-centred approach to optimising and predicting drug efficacy and safety. *New approaches to drug discovery*, pages 235–260.

Isabel Segura Bedmar, Paloma Martínez, and María Herrero Zazo. 2013. Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013). Association for Computational Linguistics.

Irena Spasić, Evangelos Simeonidis, Hanan L. Messiha, Norman W. Paton, and Douglas B. Kell. 2009. KiPar, a tool for systematic information retrieval regarding parameters for kinetic modelling of yeast metabolic pathways. *Bioinformatics*, 25(11):1404–1411.

Mujeen Sung, Minbyul Jeong, Yonghwa Choi, Donghyeon Kim, Jinhyuk Lee, and Jaewoo Kang. 2022. Bern2: an advanced neural biomedical named entity recognition and normalization tool. *arXiv preprint arXiv:2201.02080*.

Bruno Taillé, Vincent Guigue, Geoffrey Scoutheeten, and Patrick Gallinari. 2020. Let's stop incorrect comparisons in end-to-end relation extraction! *arXiv preprint arXiv:2009.10684*.

Achakulvisut Titipat and Daniel Acuna. 2015. Pubmed Parser: A Python Parser for PubMed Open-Access XML Subset and MEDLINE XML Dataset.

Jyh Jong Tsay, Bo Liang Wu, and Chang Ching Hsieh. 2009. Automatic extraction of kinetic information from biochemical literatures. *6th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2009*, 5:28–32.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. *arXiv preprint arXiv:1909.03546*.

Yuchen Wang, Haichun Liu, Yuanrong Fan, Xingye Chen, Yan Yang, Lu Zhu, Junnan Zhao, Yadong Chen, and Yanmin Zhang. 2019. In silico prediction of human intravenous pharmacokinetic parameters with improved accuracy. *Journal of chemical information and modeling*, 59(9):3968–3980.

Zhiping Wang, Seongho Kim, Sara K Quinney, Yingying Guo, Stephen D Hall, Luis M Rocha, and Lang Li. 2009. Literature mining on pharmacokinetics numerical data: a feasibility study. *Journal of biomedical informatics*, 42(4):726–735.

Chi Heem Wong, Kien Wei Siah, and Andrew W Lo. 2019. Estimation of clinical trial success rates and related parameters. *Biostatistics*, 20(2):273–286.

Zexuan Zhong and Danqi Chen. 2020. A frustratingly easy approach for entity and relation extraction. *arXiv preprint arXiv:2010.12812*.

A Appendix: Corpus Construction

A.1 Annotation Guidelines

The annotation guidelines for annotating entities and relations of PK estimations from scientific sentences can be found at `https://github.com/PKPDAI/PKRelations/blob/master/docs/Annotation_Guidelines_PKRelations.pdf`. Annotators were asked to base their labelling decisions on these guidelines, which were updated accordingly as new cases appeared.

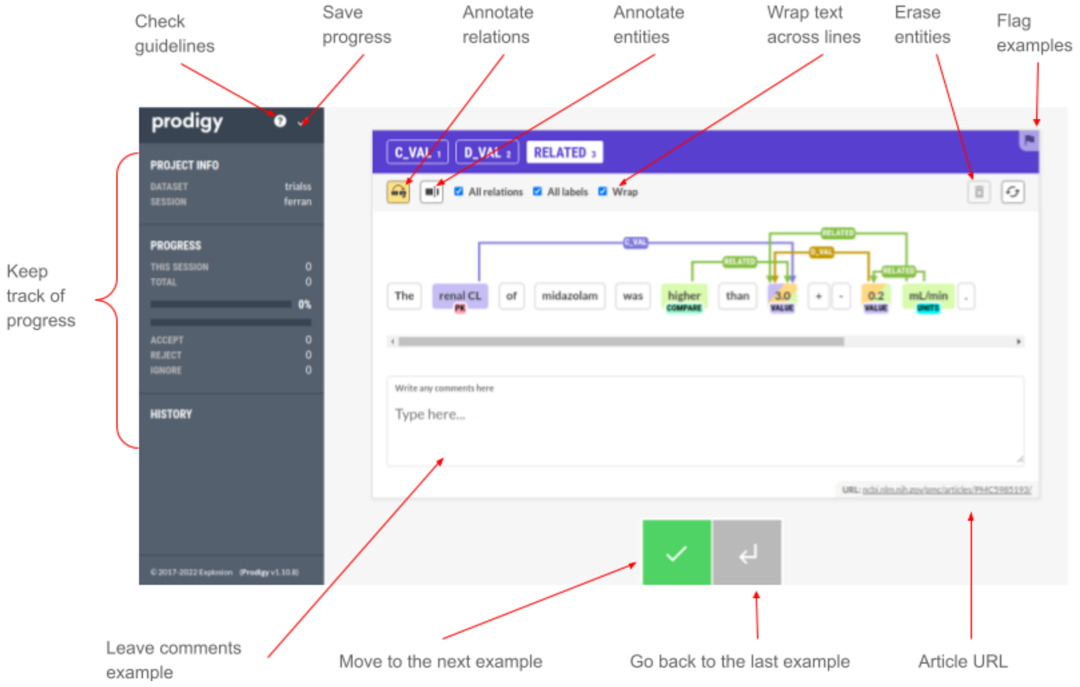
Final Annotation Check. After multiple expert annotators had annotated the development and test sets, a final check involved comparing model predictions against their annotated version. This allowed for identifying potentially missed entities and relations during the annotation.

A.2 Annotation Interface

The annotation interface (see Figure 3) was developed in Prodigy (Montani, Ines and Honnibal, 2018) and allowed annotation of both entities and relations at the sentence level. The annotators were presented with a single sentence at a time and could swap between the entity and relation annotation modes. The annotations of named entities were represented at the character level, and relations were defined with the unique identifiers of each entity and their relation class. Candidate values and ranges were pre-highlighted in the interface using a rule-based system. PK terms were pre-highlighted using the NER model from Hernandez et al. (2024), and a list of dictionary terms was used to pre-annotate Compare entities.

A.3 Corpus Limitations

The main limitation of PRED is the potential bias in selecting candidate sentences. The sampled sentences went through two filtering stages that involved model predictions: (1) selection of PK-relevant documents identified by the document classifier from Hernandez et al. (2021) and (2) selection of sentences that at least had one PK entity recognised by our PK RE model. As a result, if the document classifier missed specific types of documents, these would not appear on this dataset. Using the trained PK NER model from Hernandez



DATASET trialss
SESSION ferran
THIS SESSION 0
TOTAL 0
0%
ACCEPT 0
REJECT 0
IGNORE 0

Write any comments here
Type here...

URL: ncbi.nlm.nih.gov/pmc/articles/PMC5985193/

© 2017-2022 Explosion Prodigy v1.10.8

Figure 3: Screenshot of the interface used to annotate entities and relations from scientific text. The example displays a single sentence after entities and relations were annotated.

et al. (2024) for filtering instances with PK parameter mentions might exclude sentences where the NER model missed a single PK mention. Furthermore, if a specific sentence mentioned more than one parameter and only one match (partial or not) was detected by the NER model, the sentence was included in the candidate pool, and these incorrect predictions were later corrected during the annotation process. Overall, it is important to consider that training RE models on this dataset and directly applying them to sentences in the literature without additional filtering might result in the extraction of non-PK measurements due to the filtering approach performed in the sampling stage. For this reason, when deploying systems in production, it is important to combine models trained on this dataset with filtering approaches to discard irrelevant measurements (e.g. pre-tagging PK parameters or posterior EL of PK mentions recognised with RE models).

KG-Rank: Enhancing Large Language Models for Medical QA with Knowledge Graphs and Ranking Techniques

Rui Yang[1]*, Haoran Liu[2], Edison Marrese-Taylor [3], Qingcheng Zeng [4], Yu He Ke[5], Wanxin Li[6],
Lechao Cheng[7], Qingyu Chen[8,9], James Caverlee[2], Yutaka Matsuo[3], Irene Li[3]*

[1]Duke-NUS Medical School, [2]Texas A&M University, [3]The University of Tokyo,
[4]Northwestern University, [5]Singapore General Hospital, [6]Zhejiang University,
[7]Zhejiang Lab, [8]Yale University, [9]National Institutes of Health
yang.rui@duke-nus.edu.sg, ireneli@ds.itc.u-tokyo.ac.jp

Abstract

Large language models (LLMs) have demonstrated impressive generative capabilities with the potential to innovate in medicine. However, the application of LLMs in real clinical settings remains challenging due to the lack of factual consistency in the generated content. In this work, we develop an augmented LLM framework, KG-Rank, which leverages a medical knowledge graph (KG) along with ranking and re-ranking techniques, to improve the factuality of long-form question answering (QA) in the medical domain. Specifically, when receiving a question, KG-Rank automatically identifies medical entities within the question and retrieves the related triples from the medical KG to gather factual information. Subsequently, KG-Rank innovatively applies multiple ranking techniques to refine the ordering of these triples, providing more relevant and precise information for LLM inference. To the best of our knowledge, KG-Rank is the first application of KG combined with ranking models in medical QA specifically for generating long answers. Evaluation on four selected medical QA datasets demonstrates that KG-Rank achieves an improvement of over 18% in ROUGE-L score. Additionally, we extend KG-Rank to open domains, including law, business, music, and history, where it realizes a 14% improvement in ROUGE-L score, indicating the effectiveness and great potential of KG-Rank.

1 Introduction

Large language models (LLMs), such as GPT-4 (OpenAI, 2023) and LLaMa2 (Touvron et al., 2023), have demonstrated powerful generative capabilities (Gao et al., 2023; Yang et al., 2024b). Despite their considerable potential in various domains, including medicine (Li et al., 2022a; Yang et al., 2023c; Ke et al., 2024; Yang et al., 2024a), their limited training on medical data raises concerns about the consistency of the generated content with established medical facts (Yang et al., 2023b; Bi et al., 2024).

To address this challenge without additional computational cost, previous research, such as Almanac (Hiesinger et al., 2023) and ChatENT (Long et al., 2023), leverages external medical knowledge to enhance the accuracy and reliability of LLM-generated content. However, merely retrieving external knowledge risks introducing irrelevant or unreliable information (Yang et al., 2024a), which can compromise the effectiveness of LLMs, and raise issues of credibility, data consistency, privacy, security, and legality. While previous studies have emphasized the advantages of utilizing external knowledge, they have overlooked a crucial question: *How to better integrate external knowledge?*

In this work, we propose **KG-Rank**, an augmented framework that integrates a structured medical knowledge graph (KG) with ranking techniques into LLMs to achieve more accurate and reliable long-form medical question-answering (QA). We first retrieve one-hop relations of related medical entities from the medical KG (Unified Medical Language System (UMLS)) (Bodenreider, 2004). To retain relevant information from the KG, we then propose to apply ranking and re-ranking methods to optimize the ordering of triplets.

Specifically, we introduce three ranking techniques to improve the integration of LLM with KG by filtering irrelevant data, highlighting key information, and ensuring diversity. These techniques also streamline the process by reducing the number of triplets required for LLM inference. Additionally, we apply re-ranking models to reassess and emphasize the most relevant triplets, enhancing the factuality of KG-Rank in the long-form medical QA task.

To summarize, our contributions are: (1) We propose KG-Rank, a KG-augmented LLM framework for the medical QA task. To the best of our knowledge, this is the first application of KG com-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 155–166
August 16, 2024. ©2024 Association for Computational Linguistics

bined with ranking techniques to enhance LLMs for medical QA with long answers. (2) We incorporate different ranking and re-ranking techniques to eliminate noise and redundancy in the KG-retrieval stage. (3) We validate the effectiveness of KG-Rank on both medical and various open-domain QA tasks. All the data and code can be found at https://github.com/YangRui525/KG-Rank.

2 Methodology

As shown in Fig. 1, we introduce the KG-Rank (Knowledge Graph -Rank) framework for the long-form medical QA task.

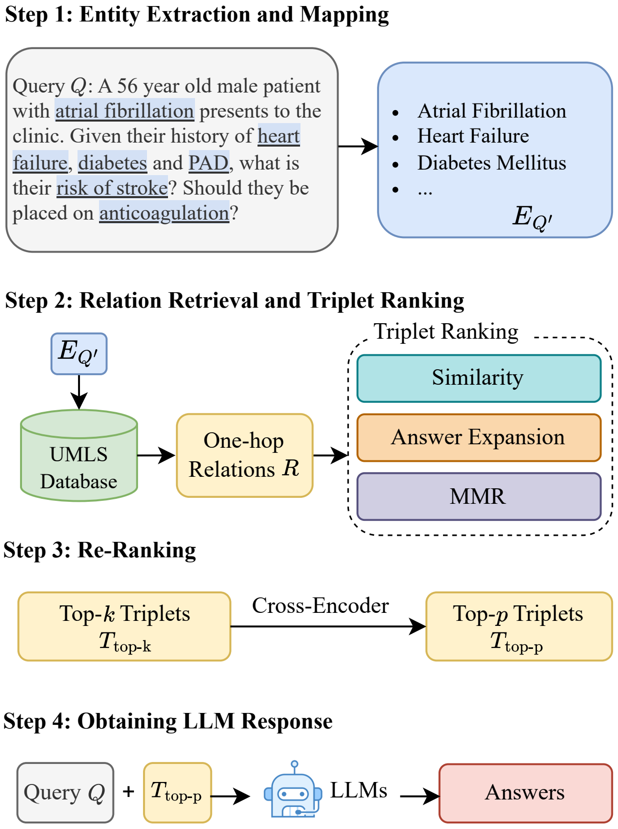

Step 1: Entity Extraction and Mapping
Query Q: A 56 year old male patient with atrial fibrillation presents to the clinic. Given their history of heart failure, diabetes and PAD, what is their risk of stroke? Should they be placed on anticoagulation?
Atrial Fibrillation
Heart Failure
Diabetes Mellitus
...
$E_{Q'}$
Step 2: Relation Retrieval and Triplet Ranking
$E_{Q'}$
UMLS Database
One-hop Relations R
Triplet Ranking
Similarity
Answer Expansion
MMR
Step 3: Re-Ranking
Top-k Triplets $T_{\text{top-k}}$
Cross-Encoder
Top-p Triplets $T_{\text{top-p}}$
Step 4: Obtaining LLM Response
Query Q
+
$T_{\text{top-p}}$
LLMs
Answers

Figure 1: An illustration of KG-Rank Framework.

2.1 External Knowledge Graph

We define the external KG as $G = (V, E)$, where V represents the set of entities and E represents the set of structural relations. For the medical QA task, we choose UMLS as the primary medical KG. UMLS is a comprehensive repository of health and biomedical vocabularies, designed to promote information standardization and interoperability. The core component of UMLS, the Metathesaurus, contains over 3.8 million concepts and more than 78 million relations, and supports 25 languages, providing extensive medical knowledge coverage to enhance LLMs. In UMLS, knowledge is represented in the form of triples, which consist of two medical concepts and the relation between them. For example, in the triple *(Myopia, clinically_associated_with, HYPERGLYCEMIA)*, "Myopia" and "HYPERGLYCEMIA" are medical concepts, while "clinically_associated_with" is the relation between them.

2.2 Entity Extraction and Mapping

In the first step, we extract key entities and find mappings from the external KG. Specifically, for the given question Q, we apply a Medical NER Prompt P_{MedNER} to identify related medical entities E_Q, and then we map each entity $e_i \in E_Q$ to the corresponding entity in the knowledge graph G. The detailed prompt can be found in Appendix A.1.

2.3 Relation Retrieval and Triplet Ranking

After identifying the corresponding entities $E_{Q'}$, we retrieve their one-hop relations from the KG (denoted as *UMLS*):

$$E_{Q'} = \{e'_i \in V \mid \exists e_i \in E_Q, e_i \mapsto e'_i\}.$$

Within UMLS, there exists extensive relational information, where one entity may be associated with thousands of one-hop relations. Consequently, to facilitate the extraction of the most relevant, we propose ranking methods. We encode the question Q and each triplet (e'_i, r, e'_j) into $\mathbf{q}, \mathbf{r}_{ij}$ through UmlsBERT (Michalopoulos et al., 2021). Then, we explore three techniques for ranking the triplets:

Similarity Ranking We compute the similarity score between the question embedding $\mathbf{q}$ and each relation embedding $\mathbf{r}_{ij}$.

Answer Expansion Ranking We first utilize LLMs to generate a hallucinatory answer A for the question Q , and then we encode the concatenation of $[Q, A]$ to obtain text embedding $\mathbf{t}$. Subsequently, we utilize the expanded question embedding $\mathbf{t}$ to search for the most similar triplets in vector space. The detailed prompt for answer expansion can be found in Appendix A.2.

MMR Ranking This method is inspired by an information extraction method Maximal Marginal Relevance (MMR) (Carbonell and Goldstein-Stewart, 1998). Initially, we identify the triplet with the highest similarity score to the question Q. For the remaining triplets, we dynamically adjust their similarity scores based on the ones that have already been selected. In this way, we could consider both relevancy and redundancy:

$$w = w_{base} + \delta \cdot n,$$

$$\text{score}_{ij} = \text{sim}(\mathbf{q}, \mathbf{r}_{ij}) - w \cdot \overline{\text{sim}}(\mathbf{r}_{ij}, \mathbf{r}_{sel}).$$

Where, w is an adjustable weight, with a base weight and δ as the incremental weight factor per selected triplet, n is the count of triplets that have been selected.

Re-ranking After the ranking stage, we obtain an ordering of the triplets. We then employ a medical cross-encoder model, MedCPT (Jin et al., 2023), to re-rank them, ensuring that the most relevant triples are chosen. The re-ranked top-p triplets, combined with the task prompt, are input into LLMs for answer generation. The detailed prompt can be found in Appendix A.3.

3 Experiments

We conduct experiments on four selected medical QA datasets, in which the answers are free-text, as shown in Tab. 1. LiveQA (Abacha et al., 2017) consists of health questions submitted by consumers to the National Library of Medicine. It includes a training set with 634 QA pairs and a test set comprising 104 QA pairs, which is used for evaluation. ExpertQA (Malaviya et al., 2023) is a high-quality long-form QA dataset with 2177 questions spanning 32 fields, along with answers verified by domain experts. Among them, 504 medical questions (Med) and 96 biology (Bio) questions were used for evaluation. MedicationQA (Abacha et al., 2019) includes 690 drug-related consumer questions along with information retrieved from reliable websites and scientific papers. We evaluate the generated answers using ROUGE (Lin, 2004), BERTScore (Zhang et al., 2019), MoverScore (Zhao et al., 2019) and BLEURT (Sellam et al., 2020).

Dataset	Sentence (Q)	Word (Q)	Sentence (A)	Word (A)
LiveQA	1.15	14.76	6.96	141.02
ExpertQA-Bio	1.26	21.69	6.18	184.38
ExpertQA-Med	1.37	22.19	5.96	180.55
MedQA	1.02	7.36	3.38	71.48

Table 1: Statistics on the average number of sentences and words across four medical datasets (Q: Question, A: Answer).

3.1 Results

As shown in Tab. 2, we evaluate GPT-4 and LLaMa2-13b across the following settings: zero-shot (ZS), and three proposed ranking techniques: Similarity Ranking (Sim), Answer Expansion Ranking (AE), and Maximal Marginal Relevance Ranking (MMR). Also with the Re-ranking (RR), which is on top of the Similarity Ranking.

3.2 Datasets

The results show that incorporating the knowledge graph and ranking techniques notably enhances performance in almost all benchmarks and evaluation metrics in the zero-shot setting, demonstrating the effectiveness of KG-Rank. Significantly, the RR method excels in the ExpertQA-Bio, ExpertQA-Med, and Medication QA datasets, particularly evident in the over 18% increase in the ROUGE-L score for ExpertQA-Bio. While KG-Rank still shows effectiveness on LiveQA, the RR method does not show steady improvement compared to other ranking techniques. This inconsistency may arise since the answers in LiveQA are generated via automatic extraction methods, leading to issues with semantic coherence and disorganized formats. Moreover, the performance of the three ranking methodologies exhibited variability across various datasets, indicating their unique strengths and limitations in differing contexts.

In assessing model performance, GPT-4 consistently surpasses LLaMa2-13b in both zero-shot and various ranking settings. Additionally, we evaluate the zero-shot performance of a medical LLM on these datasets in Section 4 (Medical LLM).

4 Ablation Study and Analysis

Medical LLM To further investigate the capability of the medical LLM, we compare the zero-shot performance of LLaMa2-7b and baize-healthcare (Xu et al., 2023) without KG-Rank. Baize-healthcare, which is fine-tuned on LLaMa-7b using medical data, consistently outperforms LLaMa2-7b across all four datasets, as shown in Fig. 2. More comparison results can be found in Appendix B.1.

Re-ranking Models We employ GPT-4 with similarity ranking as the final setting and compare two re-ranking models: the MedCPT cross-encoder model, trained on the extensive PubMed articles, and the Cohere (`https://cohere.com`) re-ranking model, designed for broader domain applications. As shown in Tab. 3, MedCPT steadily outperforms the Cohere re-rank model on all datasets, highlighting the importance of specialized re-rank models

Dataset	**Method**	**GPT-4**				**LLaMA2-13b**			
		ROUGE-L	BERTScore	MoverScore	BLEURT	ROUGE-L	BERTScore	MoverScore	BLEURT
LiveQA	ZS	18.89	82.50	54.02	39.84	17.73	81.93	53.37	40.45
	Sim	19.35	**83.01**	54.08	40.47	18.52	82.78	**53.79**	**40.59**
	AE	19.24	82.95	54.04	40.15	18.45	82.60	53.70	39.80
	MMR	19.32	82.91	54.03	**40.55**	18.25	82.70	53.67	40.22
	RR	**19.44**	82.94	**54.11**	40.50	**18.83**	**82.79**	53.72	39.59
ExpertQA-Bio	ZS	23.00	84.50	56.15	44.53	23.26	84.38	55.58	44.65
	Sim	25.90	85.72	56.73	45.10	24.96	84.91	55.83	44.35
	AE	26.78	85.77	56.79	45.18	24.84	84.97	55.72	43.55
	MMR	26.54	85.76	56.77	44.93	25.40	85.08	55.98	44.04
	RR	**27.20**	**85.83**	**57.11**	**45.91**	**25.79**	**85.18**	**56.17**	**45.20**
ExpertQA-Med	ZS	25.45	85.11	56.50	45.98	24.86	84.89	55.74	46.32
	Sim	27.61	86.10	57.13	46.47	26.40	85.50	56.23	46.15
	AE	27.98	86.12	57.25	46.80	26.15	85.36	56.17	46.02
	MMR	27.78	86.22	57.28	46.84	26.42	85.57	56.24	46.41
	RR	**28.08**	**86.30**	**57.32**	**47.00**	**27.49**	**85.80**	**56.58**	**46.47**
MedicationQA	ZS	14.41	82.55	52.62	37.41	13.30	81.81	51.96	38.30
	Sim	16.05	83.56	53.23	37.60	14.60	82.73	52.47	38.38
	AE	16.13	83.46	53.23	37.87	14.19	82.50	52.33	37.90
	MMR	15.89	83.48	53.22	37.73	14.56	82.69	52.44	38.31
	RR	**16.19**	**83.59**	**53.30**	**37.91**	**14.71**	**82.79**	**52.59**	**38.42**

Table 2: Automatic evaluation scores: we compare ROUGE-L, BERTScore, MoverScore, BLEURT on different settings. The superior scores among the same models are highlighted in **bold**.

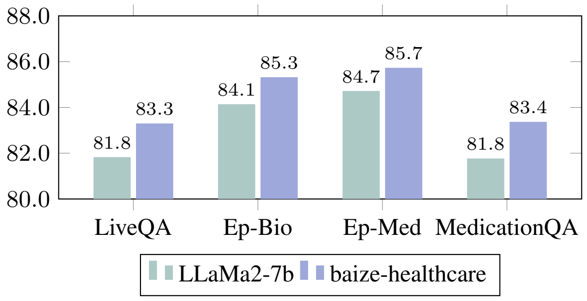


Figure 2: BERTScore comparison: zero-shot setting with LLaMa2-7b and Baize-Healthcare. Ep stands for ExpertQA.

in the medical field. Additional evaluations are provided in Appendix B.2.

Dataset	ROUGE-L	BERTScore	MoverScore	BLEURT
Cohere				
LiveQA	18.72	82.94	54.08	40.07
ExpertQA-Bio	26.08	85.81	56.93	45.70
ExpertQA-Med	27.59	86.08	57.14	46.54
MedicationQA	16.14	83.46	53.25	37.82
MedCPT				
LiveQA	**19.44**	**82.95**	**54.11**	**40.50**
ExpertQA-Bio	**27.20**	**85.83**	**57.11**	**45.91**
ExpertQA-Med	**28.08**	**86.30**	**57.32**	**46.84**
MedicationQA	**16.19**	**83.59**	**53.30**	**37.91**

Table 3: The performance of Cohere re-rank model and MedCPT in the re-ranking stage.

Case Study To further analyze the generated content of the KG-Rank framework, a case study is presented in Fig. 3. When asked about ideal diet recommendations for a 53-year-old male with acute renal failure and hepatic failure, both provide guidelines regarding protein intake. However, the original recommendation emphasizes ensuring adequate protein consumption (*1.6-2.2 grams per kilogram*), whereas the answer generated under the KG-Rank framework advises controlling protein intake (*limited to about 0.8-1 gram per kilogram*). The difference is critical for patients with acute renal and hepatic failure, where an inappropriate protein dosage, such as the higher range of 1.6-2.2 grams per kilogram, could worsen the strain on already compromised kidneys and liver, potentially leading to escalated health issues. This case shows that KG-Rank is more factually correct in the generated answer. More case studies can be found in the Appendix C.

Question: What would be the ideal diet recommendations for a 53 year old male with acute renal failure and hepatic failure?

LLaMa2-13b: ...3. Protein intake: Provide adequate protein to support liver function and wound healing. The recommended daily intake of protein for a 53-year-old male is approximately 1.6-2.2 grams per kilogram of body weight...

LLaMa2-13b w KG-Rank:...1. Low protein intake: Protein intake should be limited to about 0.8-1.0 gram per kilogram of body weight per day, as excessive protein intake can exacerbate renal failure and liver disease...

Figure 3: A case study from ExpertQA-Med: results from LLaMa2-13b and with KG-Rank.

LLM-based Evaluation Although KG-Rank achieves significant improvements in ROUGE, BERTScore, MoverScore, and BLEURT, these automatic scores may have limitations in evaluating the factuality of long-form medical QA. Therefore, we introduce GPT-4 score specifically for factuality evaluation (Zheng et al., 2024). The evaluation criteria are designed by two resident physicians with over five years of experience, which can be found in Appendix A.4. As shown in Tab. 4, we choose GPT-4 as the vanilla model, and KG-Rank outperforms the zero-shot setting across all datasets.

Dataset	Zero-Shot	Tie	KG-Rank
LiveQA	0	43	**61**
ExpertQA-Bio	0	43	**52**
ExpertQA-Med	3	235	**266**
MedQA	8	211	**468**

Table 4: GPT-4 evaluation across four medical datasets.

KG-Rank in Open Domain Additionally, to demonstrate the effectiveness of our KG-Rank, we extend it to the open domain by replacing UMLS with Wikipedia through the DBpedia API (`https://www.dbpedia.org/`). We conduct the experiment on Mintaka (Sen et al., 2022), which is a complex, natural, and multilingual dataset designed for experimenting with end-to-end question-answering models. We randomly select 1,000 pairs from the test set for evaluation. Under the enhancement of the KG-Rank framework, the accuracy increases from 60.40% to 61.90%. The detailed prompt can be found in Appendix A.5.

We also conduct experiments in the domains of law, business, music, and history using the ExpertQA dataset. We employ GPT-4 as the vanilla model and use ROUGE-L, BERTScore, and MoverScore for evaluation. As shown in Tab. 5, KG-Rank outperforms the baseline across all benchmarks. Building on these findings, the effectiveness of our framework is not limited to the medical domain but can also be applied to various other fields. For more case studies, please refer to Appendix C.

Setting	ROUGE-L	BERTScore	MoverScore
ExpertQA-Law			
Base	26.33	85.03	48.57
KG-Rank	**29.93**	**86.25**	**48.63**
ExpertQA-Business			
Base	21.78	84.46	48.92
KG-Rank	**24.20**	**85.42**	**49.10**
ExpertQA-Music			
Base	23.84	85.21	45.73
KG-Rank	**27.31**	**86.23**	**46.55**
ExpertQA-History			
Base	25.65	85.55	45.82
KG-Rank	**27.75**	**86.21**	**47.08**

Table 5: Base and KG-Rank performance in the open domain.

5 Conclusion

In this work, we propose KG-Rank, an enhanced LLM framework that integrates a medical KG and ranking techniques to improve the factuality of medical QA. As far as we know, KG-Rank is the first application of KG combined with ranking techniques for long-answer medical QA. Across four medical QA datasets, KG-Rank demonstrates over an 18% improvement in ROUGE-L score. Its application to open domains yields a 14% ROUGE-L score enhancement, underscoring KG-Rank's effectiveness and versatility.

Limitations

In this research, we propose an LLM framework augmented by UMLS to improve the quality of the content generated. However, there are some limitations, which we will address in the next phase. Firstly, we plan to incorporate physician evaluations to validate the factual accuracy of KG-Rank's answers. Secondly, we aim to assess the performance of more medical-specific base models on medical QA tasks. Lastly, while the ranking method may increase computational time, we recognize the need to optimize its efficiency. We will consider graph-based methods (Yang et al., 2023a; Li et al., 2022b) and some efficiency methods (Feng et al., 2023).

Ethical Considerations

This research utilize public medical datasets solely for academic purposes, not for practical application. We employ GPT-4, LLaMa2-13b, LLaMa2-7b, baize-healthcare for text generation, ensuring that no harmful content was produced. Both the benchmark datasets and the model outputs are free of any individual privacy data.

References

Asma Ben Abacha, Eugene Agichtein, Yuval Pinter, and Dina Demner-Fushman. 2017. Overview of the medical question answering task at trec 2017 liveqa. In *TREC*, pages 1–12.

Asma Ben Abacha, Yassine Mrabet, Mark Sharp, Travis R Goodwin, Sonya E Shooshan, and Dina Demner-Fushman. 2019. Bridging the gap between consumers' medication questions and trusted answers. In *MedInfo*, pages 25–29.

Baolong Bi, Shenghua Liu, Lingrui Mei, Yiwei Wang, Pengliang Ji, and Xueqi Cheng. 2024. Decoding by contrasting knowledge: Enhancing llms' confidence on edited facts.

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270.

Jaime G. Carbonell and Jade Goldstein-Stewart. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Aosong Feng, Irene Li, Yuang Jiang, and Rex Ying. 2023. Diffuser: Efficient transformers with multi-hop attention diffusion for long sequences. In *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 12772–12780. AAAI Press.

Fan Gao, Hang Jiang, Moritz Blum, Jinghui Lu, Yuang Jiang, and Irene Li. 2023. Large language models on wikipedia-style survey generation: an evaluation in nlp concepts. *arXiv preprint arXiv:2308.10410*.

William Hiesinger, Cyril Zakka, Akash Chaurasia, Rohan Shad, Alex Dalal, Jennifer Kim, Michael Moor, Kevin Alexander, Euan Ashley, Jack Boyd, et al. 2023. Almanac: Retrieval-augmented language models for clinical medicine.

Qiao Jin, Won Kim, Qingyu Chen, Donald C Comeau, Lana Yeganova, W John Wilbur, and Zhiyong Lu. 2023. Medcpt: Contrastive pre-trained transformers with large-scale pubmed search logs for zero-shot biomedical information retrieval. *Bioinformatics*, 39(11):btad651.

Yu He Ke, Rui Yang, Sui An Lie, Taylor Xin Yi Lim, Hairil Rizal Abdullah, Daniel Shu Wei Ting, and Nan Liu. 2024. Enhancing diagnostic accuracy through multi-agent conversations: Using large language models to mitigate cognitive bias.

Irene Li, Jessica Pan, Jeremy Goldwasser, Neha Verma, Wai Pan Wong, Muhammed Yavuz Nuzumlalı, Benjamin Rosand, Yixin Li, Matthew Zhang, David Chang, et al. 2022a. Neural natural language processing for unstructured data in electronic health records: A review. *Computer Science Review*, 46:100511.

Irene Li, Linfeng Song, Kun Xu, and Dong Yu. 2022b. Variational graph autoencoding as cheap supervision for AMR coreference resolution. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2790–2800. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Cai Long, Deepak Subburam, Kayle Lowe, André dos Santos, Jessica Zhang, Sang Hwang, Neil Saduka, Yoav Horev, Tao Su, David Cote, et al. 2023. Chatent: Augmented large language model for expert knowledge retrieval in otolaryngology-head and neck surgery. *medRxiv*, pages 2023–08.

Chaitanya Malaviya, Subin Lee, Sihao Chen, Elizabeth Sieber, Mark Yatskar, and Dan Roth. 2023. Expertqa: Expert-curated questions and attributed answers.

George Michalopoulos, Yuanxin Wang, Hussam Kaka, Helen Chen, and Alexander Wong. 2021. Umlsbert: Clinical domain knowledge augmentation of contextual embeddings using the unified medical language system metathesaurus.

OpenAI. 2023. Gpt-4 technical report.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. *arXiv preprint arXiv:2004.04696*.

Priyanka Sen, Alham Fikri Aji, and Amir Saffari. 2022. Mintaka: A complex, natural, and multilingual dataset for end-to-end question answering.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. *arXiv preprint arXiv:2304.01196*.

Boming Yang, Dairui Liu, Toyotaro Suzumura, Ruihai Dong, and Irene Li. 2023a. 10024 going beyond local: Global graph-enhanced personalized news recommendations. In *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*, pages 24–34. ACM.

Rui Yang, Yilin Ning, Emilia Keppo, Mingxuan Liu, Chuan Hong, Danielle S Bitterman, Jasmine Chiat Ling Ong, Daniel Shu Wei Ting, and Nan Liu. 2024a. Retrieval-augmented generation for generative artificial intelligence in medicine. *arXiv preprint arXiv:2406.12449*.

Rui Yang, Ting Fang Tan, Wei Lu, Arun James Thirunavukarasu, Daniel Shu Wei Ting, and Nan Liu. 2023b. Large language models in health care: Development, applications, and challenges. *Health Care Science*.

Rui Yang, Boming Yang, Sixun Ouyang, Tianwei She, Aosong Feng, Yuang Jiang, Freddy Lecue, Jinghui Lu, and Irene Li. 2024b. Leveraging large language models for concept graph recovery and question answering in nlp education. *arXiv preprint arXiv:2402.14293*.

Rui Yang, Qingcheng Zeng, Keen You, Yujie Qiao, Lucas Huang, Chia-Chun Hsieh, Benjamin Rosand, Jeremy Goldwasser, Amisha D Dave, Tiarnan D. L. Keenan, Emily Y Chew, Dragomir Radev, Zhiyong Lu, Hua Xu, Qingyu Chen, and Irene Li. 2023c. Ascle: A python natural language processing toolkit for medical text generation.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M Meyer, and Steffen Eger. 2019. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. *arXiv preprint arXiv:1909.02622*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

A Prompt Templates

In this section, we present the detailed prompt templates employed as inputs for LLMs at each phase of the KG-Rank process.

A.1 Medical NER Prompt

Fig. 4 illustrates the Medical NER prompt template that is specifically designed for extracting medical terminologies from a given question.

Question: {question}

You are interacting with a knowledge graph that contains definitions and relational information of medical terminologies. To provide a precise and relevant answer to this question, you are expected to:

1. Understand the Question Thoroughly: Analyze the question deeply to identify which specific medical terminologies and their interrelations, as extracted from the knowledge graph, are crucial for formulating an accurate response.
2. Extract Key Terminologies: Return the 3-5 most relevant medical terminologies based on their significance to the question.
3. Format the Output: Return in a structured JSON format with the key as "medical terminologies".

For example:
{"medical terminologies": ["term1", "term2", ...]}

Figure 4: Prompt used to extract medical terminologies.

A.2 Answer Expansion Prompt

Figure 5 illustrates the prompt template designed for our proposed answer expansion ranking strategy, as shown in step 2 of Fig. 1 and as described in Section 2.3.

Question: {question}

Provide an example answer to the given question.

Your answer is derived from a biomedical knowledge graph.

This knowledge graph encompasses a wide range of medical terminologies and elucidates the complex interconnections between these terms, supporting an in-depth and accurate response to the question.

Figure 5: Prompt for answer expansion ranking technique.

A.3 KG-Enhanced Prompt

Fig. 6 shows the prompt template to obtain final answers from LLMs, corresponding to step 4 in Fig. 1.

Answer the question in conjunction with the following content.

Context:
{context}

Patient:
{input}

Physician:

Figure 6: Prompt for obtaining KG-enhanced LLM answers.

A.4 Physician-Designed Criteria for GPT-4 Evaluation

Tab. 6 shows the criteria for evaluating medical long-form QA established by two resident physicians with over five years of experience. This critria is part of the GPT-4 evaluation prompt.

Evaluation Criteria
Factuality: The degree to which the generated text aligns with established medical facts, providing accurate explanations for further verification.
Readability: The extent to which the generated text is readily comprehensible to the user, incorporating suitable language and structure to facilitate accessibility.
Relevance: The extent to which the generated text directly addresses medical questions while encompassing a comprehensive range of pertinent information.
Completeness: The degree to which the generated text comprehensively portrays the clinical scenario or posed question, including other pertinent considerations.

Table 6: Physician-designed criteria for GPT-4 evaluation.

A.5 KG-Enhanced Prompt for Mintaka Task

Fig. 7 presents the prompt for obtaining KG-enhanced LLM answers, specially designed for the Mintaka dataset.

Here are some examples for output format:

Question: What is the seventh tallest mountain in North America?
Example Output: Mount Lucania

Question: What year was the first book of the A Song of Ice and Fire series published?
Example Output: 1996

Question: How old was Taylor Swift when she won her first Grammy?
Example Output: 20

Question: Has there ever been a Christian U.S. senator?
Example Output: Yes

Context:
{context}

Question:
{input}

Answer:

Figure 7: Prompt for obtaining KG-enhanced LLM answers, with special design for Mintaka dataset.

B Detailed Evaluation Results

B.1 Zero-shot Performance of Different LLMs

In this section, we evaluate the performance of widely-used LLMs on four medical datasets under the zero-shot setting. As shown in Tab. 7, the results indicate that GPT-4 performing better than the other LLMs.

Dataset	**Evaluation Metrics**					
	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	MoverScore	BLEURT
LLaMa2-7b						
LiveQA	18.87	3.60	17.44	81.83	53.28	39.43
ExpertQA-Bio	24.19	6.96	22.15	84.14	55.18	43.81
ExpertQA-Med	26.24	8.11	23.86	84.72	55.51	45.75
MedicationQA	14.19	2.60	13.12	81.77	51.94	37.32
baize-healthcare						
LiveQA	17.92	2.73	16.10	**83.30**	53.41	31.30
ExpertQA-Bio	23.45	6.52	21.31	**85.32**	54.95	33.80
ExpertQA-Med	24.95	7.21	22.41	**85.73**	55.12	34.52
MedicationQA	15.05	2.48	13.59	**83.37**	52.41	31.39
LLaMa2-13b						
LiveQA	19.15	3.60	17.73	81.93	53.37	**40.45**
ExpertQA-Bio	**25.33**	**7.92**	**23.26**	84.38	55.58	**44.65**
ExpertQA-Med	27.41	8.86	24.86	84.89	55.74	**46.32**
MedicationQA	14.42	2.62	13.30	81.81	51.96	**38.30**
GPT-4						
LiveQA	**20.54**	**4.65**	**18.89**	82.50	**54.02**	39.84
ExpertQA-Bio	25.06	7.84	23.00	84.50	**56.15**	44.53
ExpertQA-Med	**27.78**	**9.49**	**25.45**	85.11	**56.50**	45.98
MedicationQA	**15.52**	**3.51**	**14.41**	82.55	**52.62**	37.41

Table 7: Automatic evaluation scores: we compare ROUGE-1, ROUGE-2, ROUGE-L, BERTScore, MoverScore, BLEURT on the zero-shot setting for different LLMs with medical QA tasks. The best scores are highlighted in **bold**.

B.2 Performance of Different Re-rank Models

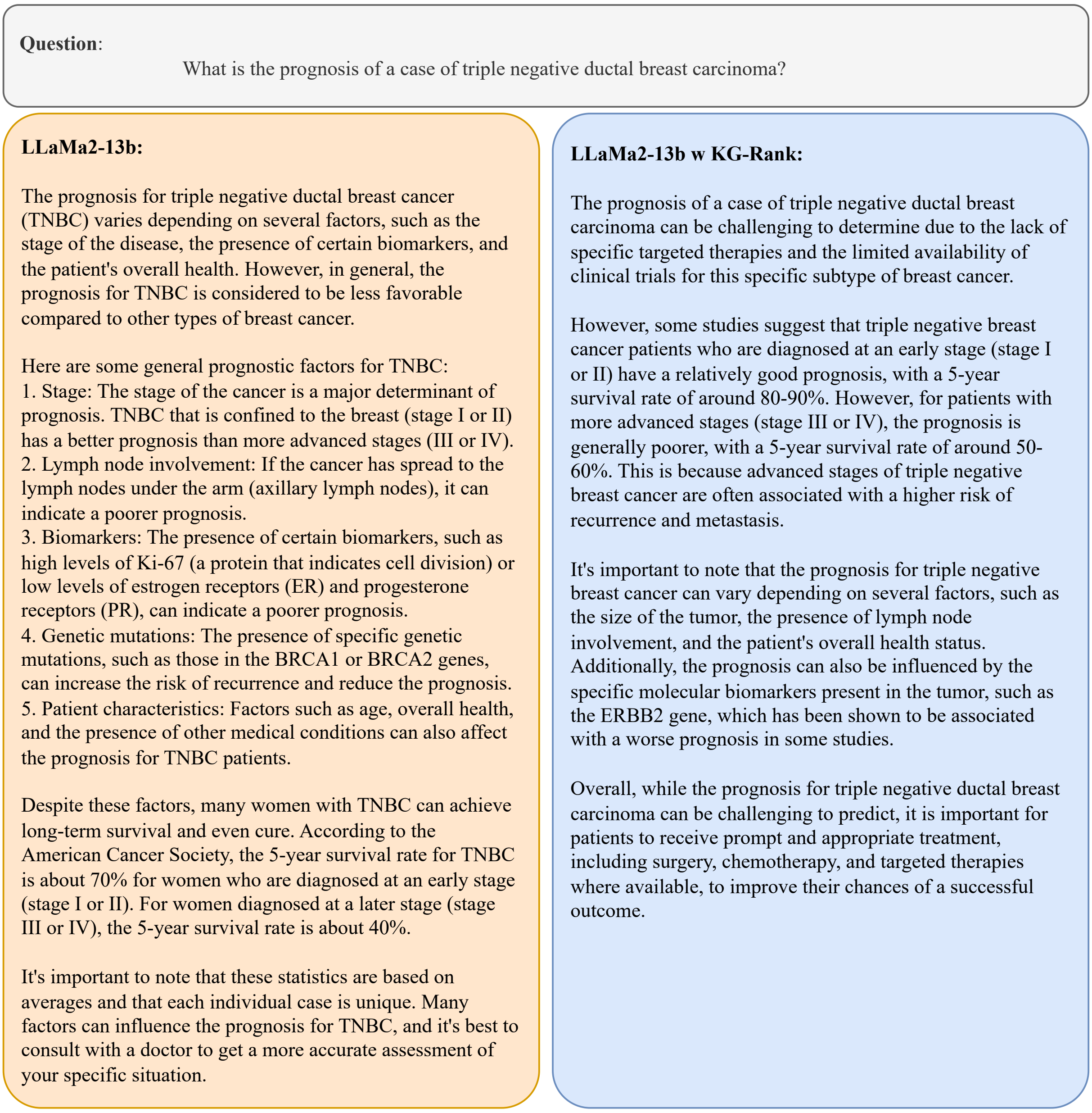
In this section, we evaluate the performance of MedCPT and the Cohere re-rank model on four medical datasets within the GPT-4 with similarity ranking setting. As shown in Table 8, the results indicate that MedCPT outperforms the Cohere re-rank model.

Dataset	**GPT-4**					
	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	MoverScore	BLEURT
Cohere						
LiveQA	21.08	4.13	18.72	82.94	54.08	40.07
ExpertQA-Bio	29.07	9.35	26.08	85.81	56.93	45.70
ExpertQA-Med	30.84	10.62	27.59	86.08	57.14	46.54
MedicationQA	17.76	3.65	16.14	83.46	53.25	37.82
MedCPT						
LiveQA	**21.70**	**4.33**	**19.44**	**82.95**	**54.11**	**40.50**
ExpertQA-Bio	**30.05**	**10.51**	**27.20**	**85.83**	**57.11**	**45.91**
ExpertQA-Med	**31.34**	**10.96**	**28.08**	**86.30**	**57.32**	**46.84**
MedicationQA	**17.94**	**3.72**	**16.19**	**83.59**	**53.30**	**37.91**

Table 8: Automatic evaluation scores: we compare the performance of different re-rank models on ROUGE-1, ROUGE-2, ROUGE-L, BERTScore, MoverScore, BLEURT. The best scores are highlighted in **bold**.

C More Case Studies

We put another case study from the ExpertQA-Med dataset, where in regards to the prognosis survival rates of breast cancer cases, the answer generated by KG-Rank is more factually accurate in terms of medical evidence, as shown in Fig. 8. Moreover, Fig. 9 shows a case study on the open-domain QA tasks from the Mintaka dataset, comparing the performance of the vanilla GPT-4 model against the KG-Rank-enhanced GPT-4 model. The case study involves a question: “How many of the Godfather movies was Robert De Niro in?” While GPT-4 responded with “2”, our proposed KG-Rank-enhanced GPT-4 provided the correct answer “1”, which matches the ground truth. We also show the evidence retrieved from DBPedia. This case study shows that by incorporating KG-Rank, the model is able to leverage the relevant information effectively to derive the correct answer, whereas the vanilla GPT-4 did not. This demonstrates the efficacy of KG-Rank in improving the accuracy of answers in LLMs when dealing with general domain factual questions.



Question:

What is the prognosis of a case of triple negative ductal breast carcinoma?

LLaMa2-13b:

The prognosis for triple negative ductal breast cancer (TNBC) varies depending on several factors, such as the stage of the disease, the presence of certain biomarkers, and the patient's overall health. However, in general, the prognosis for TNBC is considered to be less favorable compared to other types of breast cancer.

Here are some general prognostic factors for TNBC:
1. Stage: The stage of the cancer is a major determinant of prognosis. TNBC that is confined to the breast (stage I or II) has a better prognosis than more advanced stages (III or IV).
2. Lymph node involvement: If the cancer has spread to the lymph nodes under the arm (axillary lymph nodes), it can indicate a poorer prognosis.
3. Biomarkers: The presence of certain biomarkers, such as high levels of Ki-67 (a protein that indicates cell division) or low levels of estrogen receptors (ER) and progesterone receptors (PR), can indicate a poorer prognosis.
4. Genetic mutations: The presence of specific genetic mutations, such as those in the BRCA1 or BRCA2 genes, can increase the risk of recurrence and reduce the prognosis.
5. Patient characteristics: Factors such as age, overall health, and the presence of other medical conditions can also affect the prognosis for TNBC patients.

Despite these factors, many women with TNBC can achieve long-term survival and even cure. According to the American Cancer Society, the 5-year survival rate for TNBC is about 70% for women who are diagnosed at an early stage (stage I or II). For women diagnosed at a later stage (stage III or IV), the 5-year survival rate is about 40%.

It's important to note that these statistics are based on averages and that each individual case is unique. Many factors can influence the prognosis for TNBC, and it's best to consult with a doctor to get a more accurate assessment of your specific situation.

LLaMa2-13b w KG-Rank:

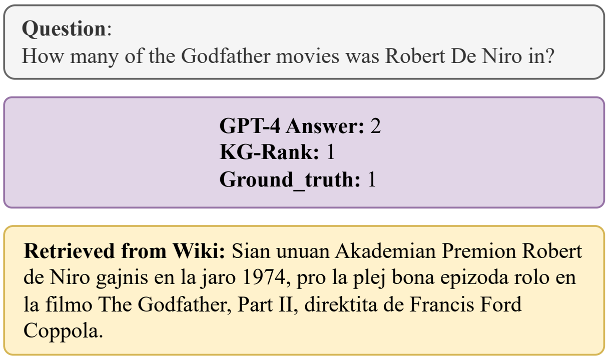
The prognosis of a case of triple negative ductal breast carcinoma can be challenging to determine due to the lack of specific targeted therapies and the limited availability of clinical trials for this specific subtype of breast cancer.

However, some studies suggest that triple negative breast cancer patients who are diagnosed at an early stage (stage I or II) have a relatively good prognosis, with a 5-year survival rate of around 80-90%. However, for patients with more advanced stages (stage III or IV), the prognosis is generally poorer, with a 5-year survival rate of around 50-60%. This is because advanced stages of triple negative breast cancer are often associated with a higher risk of recurrence and metastasis.

It's important to note that the prognosis for triple negative breast cancer can vary depending on several factors, such as the size of the tumor, the presence of lymph node involvement, and the patient's overall health status. Additionally, the prognosis can also be influenced by the specific molecular biomarkers present in the tumor, such as the ERBB2 gene, which has been shown to be associated with a worse prognosis in some studies.

Overall, while the prognosis for triple negative ductal breast carcinoma can be challenging to predict, it is important for patients to receive prompt and appropriate treatment, including surgery, chemotherapy, and targeted therapies where available, to improve their chances of a successful outcome.

Figure 8: A case study from ExpertQA-Med: we show results from vanilla LLaMa2-13b and KG-Rank-enhanced LLaMa2-13b.



Question:
How many of the Godfather movies was Robert De Niro in?

GPT-4 Answer: 2
KG-Rank: 1
Ground_truth: 1

Retrieved from Wiki: Sian unuan Akademian Premion Robert de Niro gajnis en la jaro 1974, pro la plej bona epizoda rolo en la filmo The Godfather, Part II, direktita de Francis Ford Coppola.

Figure 9: A case study from Mintaka: we show results from vanilla GPT-4 and KG-Rank-enhanced GPT-4.

D Experimental Setup

In our experimental setup, we employ `UmlsBERT`[1], `baize-healthcare`[2], `llama-2-7b-chat-hf`[3], `llama-2-13b-chat-hf`[4], `MedCPT`[5] from Hugging Face. For GPT-4, we use the OpenAI API with a zero-temperature setting. For the Cohere re-rank model, we employ it through its API. In the MMR Ranking setting, the default value for w is 0.1, and δ is set to 0.01. All experiments are conducted on a cluster equipped with 4 NVIDIA A100 GPUs. The prediction for each sample takes about a few seconds. Based on the size of each dataset, it may take up to hours to finish the evaluation.

[1] GanjinZero/UMLSBert_ENG

[2] https://huggingface.co/project-baize/baize-healthcare-lora-7B

[3] https://huggingface.co/meta-llama

[4] https://huggingface.co/meta-llama

[5] https://huggingface.co/ncbi/MedCPT-Cross-Encoder

MedExQA: Medical Question Answering Benchmark with Multiple Explanations

Yunsoo Kim, Jinge Wu, Yusuf Abdulle, Honghan Wu
Institute of Health Informatics, University College London
yunsoo.kim.23@ucl.ac.uk

Abstract

This paper introduces MedExQA, a novel benchmark in medical question-answering, to evaluate large language models' (LLMs) understanding of medical knowledge through explanations. By constructing datasets across five distinct medical specialties that are underrepresented in current datasets and further incorporating multiple explanations for each question-answer pair, we address a major gap in current medical QA benchmarks which is the absence of comprehensive assessments of LLMs' ability to generate nuanced medical explanations. Our work highlights the importance of explainability in medical LLMs, proposes an effective methodology for evaluating models beyond classification accuracy, and sheds light on one specific domain, speech language pathology, where current LLMs including GPT4 lack good understanding. Our results show generation evaluation with multiple explanations aligns better with human assessment, highlighting an opportunity for a more robust automated comprehension assessment for LLMs. To diversify open-source medical LLMs (currently mostly based on Llama2), this work also proposes a new medical model, MedPhi-2, based on Phi-2 (2.7B). The model outperformed medical LLMs based on Llama2-70B in generating explanations, showing its effectiveness in the resource-constrained medical domain. The benchmark datasets and the model can be found at https://github.com/knowlab/MedExQA.

1 Introduction

Recent advancements in large language models (LLMs) have not only enhanced their understanding of medical domain text but also improved their ability to generate coherent text with correct medical knowledge (Tu et al., 2023; Singhal et al., 2023). Chatbots, powered by these LLMs, have emerged as indispensable tools, offering unprecedented opportunities to enhance patient care, streamline clinical decision-making processes, and medical knowledge retrieval (Achiam et al., 2023; OpenAI, 2023; Groves et al., 2023). Moreover, open-source medical LLMs further enhance the usability of such technologies in hospitals by resolving the privacy concerns associated with patient data (Toma et al., 2023; Kweon et al., 2023; Chen et al., 2023).

This research in medical LLMs has been facilitated by the introduction of question-answering (QA) datasets that serve as benchmarks for evaluating the model's understanding of medical domain knowledge (Hendrycks et al., 2020; Jin et al., 2021; Pal et al., 2022; Singhal et al., 2023). The benchmark QA datasets typically consist of multiple-choice questions (MCQ), enabling researchers to readily assess the capabilities of LLMs in comprehending and responding to diverse medical inquiries. Thus, the diversity within these datasets is a key component in creating a rigorous assessment benchmark for complex medical concepts. Nonetheless, certain areas within the medical domain, such as speech language pathology, still remain uncovered by the current benchmark datasets.

As current medical QA benchmarks are often structured as MCQ, classification accuracy is used as an evaluation metric. However, classification accuracy alone may not adequately assess whether LLMs possess the nuanced medical expertise required for reasoned responses. The explanation and rationale behind the selection of a particular choice by an LLM would provide a deeper understanding of the model's capabilities and limitations in generating responses to intricate medical questions. This comprehensive evaluation, delving into the explanation and rationale, is especially important in clinical settings where misleading information such as hallucinations produced by LLMs can have serious consequences.

In order to assess the quality of the model explainability, the dataset should include a golden explanation for the reasoning behind the answer. Additionally, since there are often multiple ways to

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 167–181
August 16, 2024. ©2024 Association for Computational Linguistics

express the same rationale in text, an ideal dataset would provide a multiple set of explanations for a single QA pair. However, current benchmark datasets are not focused on providing explanations as they often lack explanations entirely or only a subset of the dataset comes with an explanation (Hendrycks et al., 2020; Jin et al., 2021; Pal et al., 2022). This limitation highlights the need for improved datasets that are explicitly designed to include comprehensive explanations.

To address this issue, this paper presents a novel QA benchmark, MedExQA, with two sets of explanations, aiming to provide a more comprehensive evaluation of LLMs in the medical domain. To diversify the knowledge coverage in the current datasets, our proposed benchmark consists of five underrepresented specialties in current datasets: biomedical engineering, clinical laboratory science, clinical psychology, occupational therapy, and speech language pathology. In this work, the datasets were used to benchmark the performance of an extensive list of LLMs, including those trained with medical domain text. With this comprehensive benchmark evaluation, we explored the effects of medical domain-specific training. Additionally, to diversify the pool of open-source medical LLMs which are currently almost all based on the Llama2 model, we introduce our own trained model, MedPhi-2, a Phi-2 model trained with medical domain text. Our MedPhi-2 model outperformed medical LLMs based on the Llama2-70B model in generating explanations for the rationale behind the answer.

The contributions of this paper are as follows:

1. **MedExQA novel datasets with explanations.** We constructed a benchmark with 5 distinct specialties within the medical domain. The datasets include two explanations for each question and answer pairs.
2. **Comprehensive Benchmark.** We evaluated an extensive list of models: 18 baseline open-source models with various sizes (from 2.7B to 70B), 3 OpenAI GPT models, as well as our model (detailed below). In terms of evaluation approach, classification accuracy, generated explanation performance, and human evaluations are considered. To highlight, this is the first benchmark using multiple explanations, and the results demonstrate that our benchmark can better evaluate language models' understanding of medical domain knowledge.
3. **MedPhi-2 model.** We trained a small language model (SLM) based on the Phi-2 model, with medical pretraining corpus and instruction-tuning datasets. The model outperformed medical LLMs based on Llama2 70B in generating explanations.
4. **Open source.** We release the datasets, model weights, and codes to facilitate the research in medical large language modeling.

2 Related Works

2.1 MMLU

MMLU (Hendrycks et al., 2020) is a benchmark designed to measure the model's ability in knowledge-intensive QA with four-way MCQs. Within the extensive list of subjects, there are nine healthcare-related subjects such as professional medicine and medical genetics. Collectively, these nine subjects comprise a total of 1,871 questions in the test set. While MMLU provides a comprehensive set of questions, it lacks explanations for the answers, thereby limiting the dataset's evaluation to mere multiple-choice classification accuracy.

2.2 MedQA

MedQA (Jin et al., 2021) is an open-ended MCQ dataset made from professional medical doctor license exams. The dataset contains questions drawn from both real exams and mock tests for the United States Medical License Exams (USMLE). 1,273 questions, each question accompanied by four or five answer choices, are provided as the test dataset. Similar to MMLU, MedQA does not include explanations for assessing the ability to generate rationale behind the answer.

2.3 MedMCQA

MedMCQA (Pal et al., 2022) is a benchmark with questions sourced from postgraduate-level Indian medical school entrance exams (AIIMS and NEET PG). The dataset covers a breadth of medical specialties, 2,400 healthcare topics and 21 subjects and provides 4,183 MCQ with four answer choices for evaluation. Although MedMCQA is known to have explanations, nearly half of the evaluation dataset lacks explanations and instances of duplicate explanations are also observed. In fact, accuracy is only reported as the evaluation metric and explanation is not used in their paper entirely. Therefore, MedMCQA is not primarily designed for the assessment of generating explanations.

Specialty	**NUM**	**ExSIM**
Biomedical Engineering	148	75.8
Clinical Laboratory Science	377	73.7
Clinical Psychology	111	79.7
Occupational Therapy	194	79.5
Speech Language Pathology	135	80.5
Total	965	78.7

Table 1: Statistics of datasets within **MedExQA**. **NUM** represents the number of questions. **ExSIM** represents the average cosine similarity of the explanation pairs.

3 MedExQA Datasets

We introduce MedExQA, a novel QA benchmark designed to tackle the limitations of existing benchmarks by incorporating two sets of explanations. This approach aims to offer a more thorough evaluation of performance in five underrepresented specialties in the medical domain: Biomedical Engineering, Clinical Laboratory Science, Clinical Psychology, Occupational Therapy, and Speech Language Pathology.

3.1 Datasets Preparation

The raw data was manually collected from diverse freely accessible online sources, including mock tests and online exams tailored to each medical professional specialty. Some questions of the mock tests and online exams have explanations for the answers, which we used the creation of the MedExQA datasets. The pass mark for the collected mock tests and online exams was 60 percent.

To ensure data integrity, rigorous preprocessing was conducted, including the removal of duplicate questions and explanations. Additionally, similar questions were identified and eliminated using BERT cosine similarity analysis (Devlin et al., 2018). Questions containing keywords specific to laws or regulations were filtered out using a manually curated list of words. Following fair use regulations[1], answer options were systematically shuffled to maintain fairness and uphold the integrity of the dataset. Furthermore, to enhance the quality and coherence of the datasets, two sets of explanations as well as the questions underwent thorough human validation. This validation process aimed to ensure that the explanations exhibited distinct writing styles and provided comprehensible reasoning for the correct answer selection.

The resulting datasets have a total of 965 questions. Table 1 provides a detailed breakdown of the number of questions for each specialty. These datasets were split into a few-shot development set and a test set. Specifically, the few-shot development set has 5 questions per specialty, while the test set consists of 940 questions in total. It is noteworthy that each subject contains a minimum of 100 test examples, a length surpassing that of most exams tailored for human assessment.

Also, to validate that each pair of explanations is different sufficiently at the individual question level, Table 1 also provides the average cosine similarity of the pairs. The overall similarity is 78.7% which indicates the lexical difference of the two corresponding versions of explanations for each question. An example of the dataset as well as the difference in the pair of example can be found in the Appendix Figure 4.

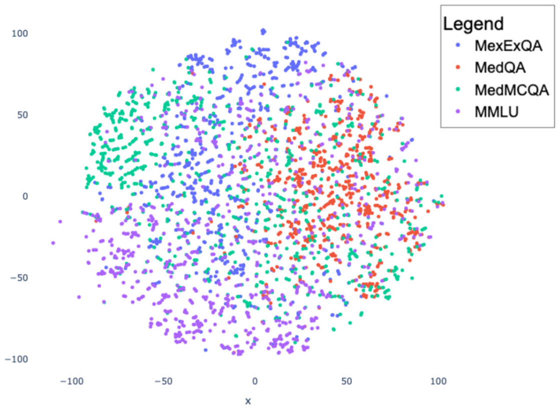


Figure 1: 2D t-SNE plot for MedExQA, MedQA, MedMCQA, and MMLU (Medicine Related 9 subjects) datasets.

3.2 Comparison of benchmark datasets

We compared MedExQA with existing benchmark datasets by visualizing their questions in the same vector space. Using t-distributed Stochastic Neighbour Embedding (tSNE), each question is represented as a point in the vector space. We used the ‘all-mpnet-base-v2’ sentence transformer model in `sklearn` package `tSNE` to retrieve vectors from questions. 965 questions were randomly sampled from each dataset. There is a cluster towards the top region mainly composed of questions from MedExQA, which clearly demonstrates its novelty compared to existing medical QA datasets.

[1] https://www.copyright.gov/fair-use/more-info.html

4 Methods

For all the experiments in this paper, both training and evaluation, we used 8 A6000 GPUs.

4.1 Baseline Models

We explored 18 baseline models with different sizes from 2.7B to 70B. Table 2 provides a comprehensive overview of the baseline models used in this paper, while more detailed descriptions of each model are available in the appendix. In cases where multiple sizes of a model are used, we distinguish each version by appending the model size to the model name. For example, the Llama2 models with sizes 7, 13, and 70B are denoted as **Llama2-7B**, **Llama2-13B**, and **Llama2-70B**, respectively. On the other hand, when a model has only one size, we refer to it solely by its name. For instance, **ClinicalCamel** denotes the ClinicalCamel 70B model.

Llama2 Variant Models	**Model Size**
Llama2(Touvron et al., 2023)	7B,13B,70B
ClinicalCamel(Toma et al., 2023)	70B
Asclepius(Kweon et al., 2023)	7B,13B
Med42(M42, 2023)	70B
AlpaCare(Zhang et al., 2023)	7B,13B
Meditron(Chen et al., 2023)	7B,70B
Medinote(Bonnet et al., 2023)	7B,13B
Other foundational models	**Model Size**
Mistral(Jiang et al., 2023)	7B
Yi(01.AI, 2024)	6B
Phi-2(Microsoft, 2023)	2.7B
SOLAR(Kim et al., 2023)	10.7B
InternLM2(Shanghai AI Lab)	7B

Table 2: Baseline Models. The models are sorted in the order of release dates.

4.2 Training MedPhi-2

As far as we know, all the publicly available open-source medical LLMs are based on Llama models, we further extended our work to test the effect of medical domain training on a different foundational model. Phi-2 model was further trained using the medical datasets that are publicly available. We pretrained Phi-2 with a 110M medical-related corpus. We further finetuned the continued pretrained model with 239K instructions. We refer to the resulting model as **MedPhi-2** throughout our paper. Table 3 summarizes the detailed composition of our training dataset. We used LLaMaFactory[2] and used Deep3 for efficient training. For both pretraining and finetuning, We trained the model with a batch size of 16 and a learning rate of 1e-5 with 3 epochs, which took 36 hours in total.

Pretrain	**Tokens**
Meditron Medical Guidelines[3]	48.3M
SNOMED CT descriptions[4]	28.3M
Biomedical Article Abstracts[5]	13.6M
Wikipedia Medical Terms[6]	13.3M
PMC Patients Notes[7]	6.7M
Finetuning	**Instructions**
Asclepius Instruction[8]	158,114
AlpaCare Instruction[9]	52,002
NHS QA and Medical Task[10]	29,354

Table 3: MedPhi-2 training data. The number of tokens for pretraining data and the number of instructions for finetuning data are listed.

4.3 Evaluation

We evaluated all models with test datasets except for human evaluation, which was performed on the development datasets. For all the evaluations, we used zero-shot, a batch size of 1, temperature of 0. To benchmark the performance of closed source models we further extended to include OpenAI's GPT models. We used `GPT3.5_1106`, `GPT4.0_1106`, and `GPT4.0_0125` APIs[11].

4.3.1 Classification Accuracy - Logits

Classification accuracy of MCQ for generative models relies on classifying the next token using logits. In other words, the token with the highest logit value is selected as the model's predicted answer. However, this approach cannot assess the model's understanding of the rationale behind the answer. We exclude GPT models for this evaluation, as we are not able to get the logit value for the next token.

[2] https://github.com/hiyouga/LLaMA-Factory

[3] https://huggingface.co/datasets/epfl-llm/guidelines

[4] https://huggingface.co/datasets/FremyCompany/AGCT-Dataset

[5] https://huggingface.co/datasets/paniniDot/sci_lay

[6] https://huggingface.co/datasets/gamino/wiki_medical_terms

[7] https://huggingface.co/datasets/zhengyun21/pmc-patients

[8] https://huggingface.co/datasets/starmpcc/asclepius-synthetic-clinical-notes

[9] https://huggingface.co/datasets/casey-martin/medinstruct

[10] https://github.com/CogStack/OpenGPT

[11] https://platform.openai.com/docs/models

4.3.2 Classification Accuracy - Chat

We utilize string-matching using regular expressions and `thefuzz` package to assess the model's proficiency in generating accurate textual responses. This approach involves searching the specific phrase for the answer choice or the choice letter within the generated response, enabling a more realistic evaluation for the model's performance.

4.4 Explanation Generation

The quality of generated explanations is further assessed using a combination of general lexical metrics. **BLEU (Papineni et al., 2002).** measures the geometric mean of precision scores of the generated explanations compared to reference explanations based on n-gram matches. **ROUGE (Lin, 2004).** assesses the similarity between generated and reference explanations, with ROUGE-L, providing a score that combines precision and recall based on the longest common subsequence. **METEOR (Banerjee and Lavie, 2005).** considers the semantic similarity and lexical variations with WordNet. **BERTScore (Zhang et al., 2019).** uses contextual embeddings, scibert embedding (Beltagy et al., 2019) for our work, to capture nuances in the semantics of the explanations. All the metrics are calculated using `evaluate` package.

We propose an enhanced methodology for evaluating models' understanding of medical domain knowledge by incorporating classification accuracy based on string matches into calculating these metrics. We assign a score of 0 to responses with incorrect answers based on string-matching classification results.

4.5 Evaluation - Human Evaluation

For human evaluation, three human annotators with MSc degrees in health-related subjects participated in assessing the quality of generated explanations. The evaluation process involved assigning a score for each explanation-answer pair based on the following rules:

1. **Score 0** the answer was incorrect, no explanation was provided, and/or the explanation is fully irrelevant.
2. **Score 0.5** the answer was correct, but the explanation or rationale was incorrect. Also, an incomplete explanation that ended with an incomplete sentence.
3. **Score 1.0** when both the answer and explanation were correct.

Although this human evaluation was performed on a small scale (development dataset: 5 samples for each specialty), this systematic evaluation process ensured a comprehensive assessment of the models' performance in providing accurate and coherent explanations.

5 Results and Discussion

5.1 Classification Accuracy - Logits

Table 4 shows the detailed results of all models. As expected, smaller language models demonstrated lower accuracy across specialties than larger models. Med42 showed the best overall performance. It showed outstanding performance in Biomedical Engineering and Clinical Laboratory Science (83.2% and 84.9% respectively). It performed on par with Meditron-70B in Clinical Psychology (84.9%). In Occupational Therapy, Llama2-70B showed the highest accuracy (80.4%). All models underperformed in Speech Language Pathology, with SOLAR performing the best (33.1%).

The effect of continued training is observed only in some models. MedPhi-2 demonstrated better performance than Phi-2, and this improvement was also found in AlpaCare-13B compared to Llama2-13B and Med42 compared to Llama2-70B. However, ClinicalCamel and Meditron-70B performed worse than Llama2-70B. This drop in performance could be due to task-specific challenges as some models may not effectively handle varied levels of specificity in MedExQA.

5.2 Classification Accuracy - Chat

Classification accuracy using chat decreased in most of the models (Table 4). Phi-2, Llama2-13B, Yi, InternLM2, and Meditron-70B did not pass the pass mark indicating these models are not robust. Meditron-70B showed the biggest performance drop by 29.3%. Llama2-70B also showed a significant performance drop in this testing by 28.5%, although it passed in Biomedical Engineering. Of the 70B models we tested, ClinicalCamel was the most robust model (7.7% decrease), and it scored higher than Med42 by 0.7%.

Our model, MedPhi-2 was the most robust model among the passed ones (0.2% decrease), and it outperformed AlpaCare-13B, Meditron-70B, Llama2-70B. This result highlights the importance of the supervised finetuning with in-domain instructions of high quality as more robust models, such as AlpaCare, ClinicalCamel, and MedPhi-2, were

Model	**BE**	**CP**	**SLP**	**OT**	**CLS**	**MAvg**
Medinote-7B	33.6 (-4.9)	34.9 (-8.5)	23.1 (6.2)	38.1 (-8.5)	44.6 (-11.6)	34.9 (-5.5)
Meditron-7B	37.8 (-7.7)	46.2 (-16.0)	20.8 (2.3)	42.9 (-10.6)	43.3 (-6.7)	38.2 (-7.8)
Llama2-7B	42.0 (-9.1)	47.2 (-9.4)	22.3 (1.5)	40.2 (-12.7	47.6 (-17.5)	39.9 (-9.4)
Asclepius-7B	44.8 (-11.2)	47.2 (-17.0)	27.7 (-1.5)	42.9 (-15.3)	45.2 (-13.4)	41.5 (-11.7)
Medinote-13B	46.2 (-18.9)	52.8 (-30.2)	28.5 (-4.6)	49.2 (-28.1)	52.4 (-20.2)	45.8 (-20.4)
AlpaCare-7B	53.2 (6.3)	53.8 (1.9)	26.9 (6.2)	59.8 (-3.7)	54.6 (-0.5)	49.6 (2.0)
Asclepius-13B	57.3 (-21.0)	56.6 (-33.0)	25.4 (-3.8)	59.8 (-34.4)	56.5 (-22.9)	51.1 (-23.0)
Phi-2	61.5(-35.7)	68.9 (-38.7)	26.2 (2.3)	64.0 (-43.4)	50 (-25.0)	54.1 (-28.1)
Llama2-13B	63.6 (-26.6)	65.1 (-42.8)	27.7 (16.2)	60.9 (-28.8)	59.4 (-17.5)	55.3 (-19.9)
MedPhi-2	65.7 (-5.6)	70.8 (0.0)	23.1 (0.0)	65.1 (-0.5)	55.1 (5.1)	56.0 (-0.2)
AlpaCare-13B	67.1 (-4.9)	69.8 (-10.4)	26.9 (-1.5)	65.1 (-4.8)	61.6 (-4.3)	58.1 (-5.2)
Mistral	75.5 (-11.2)	73.6 (-10.4)	32.3 (-6.2)	75.7 (-6.3)	71.2 (0.0)	65.7 (-6.8)
Meditron-70B	78.3 (-36.4)	84.9 (-43.4)	30.8 (-5.4)	69.8 (-37.0)	68.6 (-24.2)	66.5 (-29.3)
Yi	75.5 (-20.3)	83.0 (-28.3)	30.8 (0.8)	74.1 (-20.6)	73.4 (-17.2)	67.4 (-17.1)
SOLAR	74.8 (0.0)	81.1 (-2.8)	**33.1** (-7.7)	73.0 (-1.1)	76.1 (-3.2)	67.6 (-3.0)
InternLM2	77.6 (-25.2)	82.1 (-38.7)	29.2 (-5.4)	74.6 (-36.0)	75.0 (-33.6)	67.7 (-27.8)
ClinicalCamel	78.3 (-6.3)	84.0 (-14.1)	28.5 (-5.4)	79.9 (-6.3)	75.8 (-6.2)	69.3 (-7.7)
Llama2-70B	78.3 (-10.5)	84.0 (-47.2)	31.5 (-10.8)	80.4 (-44.4)	72.9 (-29.8)	69.4 (-28.5)
Med42	83.2 (-14.)	84.9 (-10.4)	31.5 (-4.6)	79.4 (-13.8)	80.9 (-12.6)	72.0 (-11.1)
GPT3.5_1106	72.0	82.1	29.2	70.4	71.5	65.0
GPT4_1106	86.7	86.8	31.5	88.4	**91.7**	77.0
GPT4_0125	**90.2**	**91.5**	30.8	**90.0**	**91.7**	**78.8**

Table 4: MCQ accuracy (%) using logits vs chat generation. The MCQ accuracy using logits is reported (except for GPT models). The performance gain/loss with chat generation approach is marked in parenthesis. "BE": Biomedical Engineering; "CP": Clinical Psychology; "SLP": Speech Language Pathology; "OT": Occupational Therapy; "CLS": Clinical Laboratory Science; "MAvg": Macro Average.

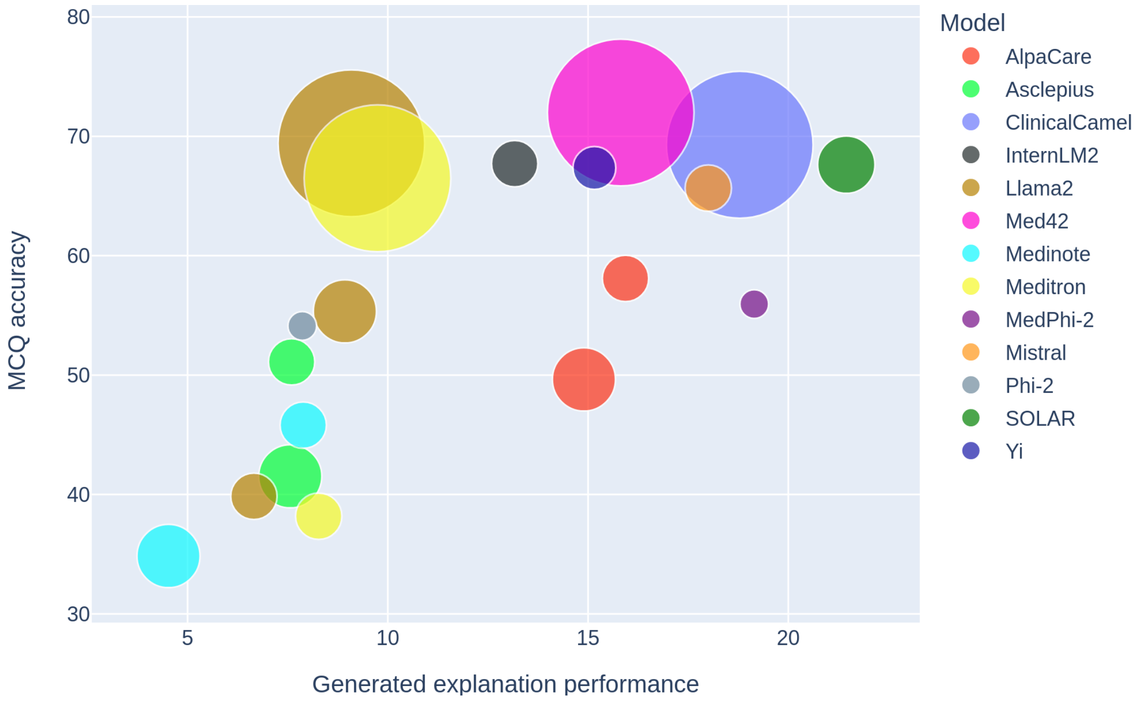


Figure 2: Scatter plot of model performance. The Y-axis is the macro average of accuracy based on logits (Table 4). The X-axis is the average score of generated explanations (Table 5). The dot size is proportional to the model size.

Model	Size (B)	ROUGE-L	METEOR	BLEU	BERTScore	AVG
Medinote	13	1.88	2.79	0.46	12.96	4.52
Llama2	7	4.92	4.03	0.16	17.52	6.66
Asclepius	13	6.12	6.12	0.32	17.70	7.56
Asclepius	7	6.07	5.61	0.22	18.48	7.60
Phi-2	2.7	5.77	7.51	1.76	16.41	7.86
Medinote	7	4.78	7.82	2.14	16.81	7.89
Meditron	7	5.15	7.96	2.56	17.43	8.27
Llama2	13	6.65	6.89	1.37	20.80	8.93
Llama2	70	6.41	6.71	1.40	21.84	9.09
Meditron	70	7.42	8.32	1.63	21.59	9.74
InternLM2	7	10.30	12.20	3.89	26.28	13.17
AlpaCare	13	11.56	11.97	2.77	33.29	14.90
Yi	6	10.97	13.25	4.79	31.62	15.16
Med42	70	11.03	12.88	3.46	35.89	15.82
AlpaCare	7	12.43	14.19	3.64	33.47	15.94
Mistral	7	12.59	17.49	5.28	36.66	18.00
ClinicalCamel	70	13.45	17.38	5.52	38.80	18.79
MedPhi-2	2.7	15.26	17.75	6.13	37.45	19.15
SOLAR	10.7	16.45	20.17	6.72	42.46	21.45
GPT3.5_1106	-	21.71	25.99	14.07	46.59	27.09
GPT4_1106	-	23.08	**35.74**	14.40	**54.50**	31.93
GPT4_0125	-	**24.83**	35.21	**16.71**	54.40	**32.79**

Table 5: Explanation Generation performance (average across the 5 subjects for each evaluation metric).

instruction-tuned with medical domain data, while Meditron-70B was just further pretrained.

GPT4_0125, GPT4_1106, and GPT3.5_1106 outperformed all the open-source models. Even with the addition of high-performing closed-source models, there is still a universal failure in performance for Speech Language Pathology.

5.3 Combining Classification Accuracy with Generated Explanation Performance

Figure 2 shows the relationship between model size and accuracy achieved in both MCQ (using logits) and generation performance. Generally, larger models tend to exhibit better performance as 70B models perform better than most of the other smaller models. However, SOLAR, Yi, and Mistral stand out as these smaller general domain models demonstrate competitive performance to the 70B medical LLMs. Further training on these foundation models holds great promise as we have seen with the Phi-2 model.

All medical LLMs with 13B (AlpaCare, Asclepius, and Meditron) exhibit worse performance in both MCQ accuracy and generation performance compared to their 7B counterparts. In fact, **Medinote-13B** is the worst-performing model. Also, 70B models do not always perform better than smaller models as Meditron-70B and Llama2-70B performed worse than many smaller models including AlpaCare and our model in the generation of reasonable explanations.

The performance evaluation presented in Table 5 also provides valuable insights into the efficacy of various models in generating explanations. Among the models evaluated, our model, **MedPhi-2** stands out in generating reasonable explanations as it outperformed all medical LLMs including 70B models. This result confirms the findings of Section 5.2 which highlighted the importance of supervised finetuning with in-domain instructions.

The SOLAR model performed the best among the open-source models, suggesting its competitive capability in explanation generation although it was not trained specifically for the medical domain. However, even this best-performing open-source model demonstrates a significant performance gap (5.64) compared to the worst-performing closed-source model, GPT3.5_1106, indicating the substantial advancements in OpenAI's GPT models.

Interestingly, despite the recent release of GPT4,

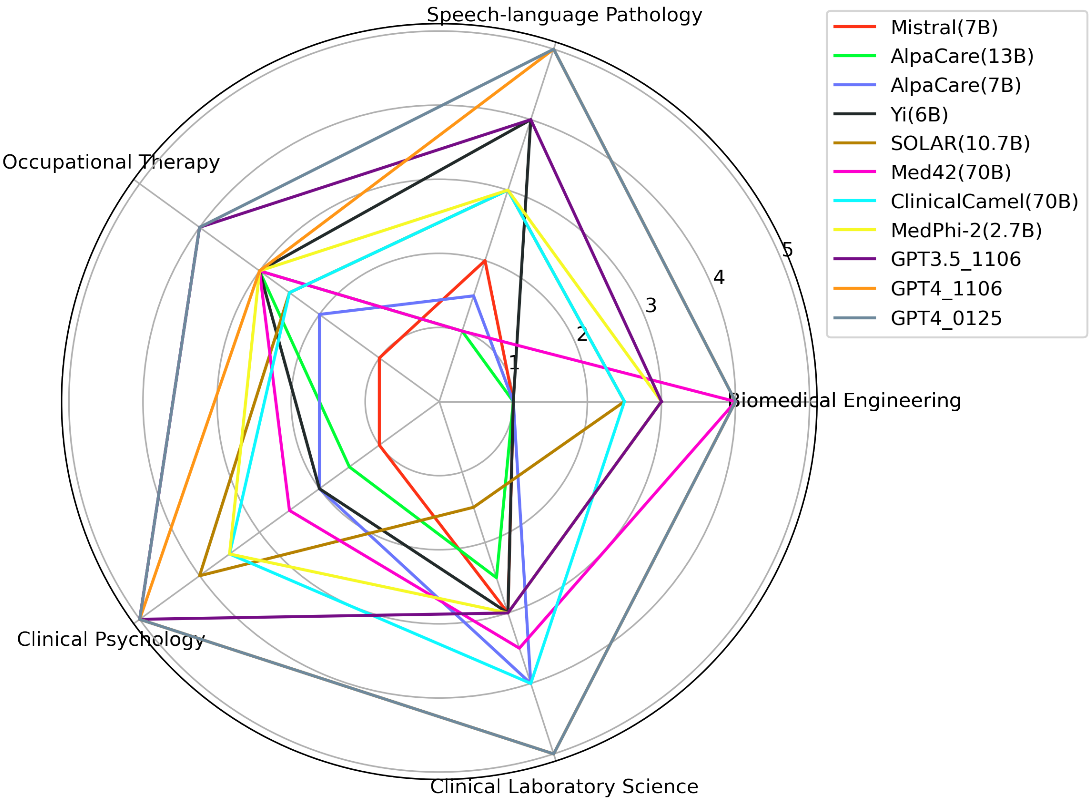


Figure 3: Human evaluation on the generated explanations, which scales from 0 to 5. The models in the legend are ordered by macro average from lowest to highest. Only models passed (3 or above) in at least one of the specialties are included.

the performance varies across different evaluation metrics. While the most recent release outperforms GPT4_1106 on average, GPT4_1106 still shows superior performance in METEOR and BERTScore. This highlights the importance of considering multiple metrics and nuances in model performance assessment, as different models may excel in distinct aspects of explanation generation.

5.4 Evaluation - Human Evaluation

Human evaluation of generated responses reveals that MedPhi-2 has the best quality among the open-source models (Figure 3). Our model was the only open-source model that passed (a score of 3 or above) in all specialties in MedExQA. In fact, **MedPhi-2** on par with **GPT3.5_1106** in Biomedical Engineering and Clinical Laboratory Science, and with **GPT4_1106** in Occupational Therapy.

The performance of models in Speech Language Pathology during human evaluation was relatively decent, which contrasts with results obtained through other evaluation methods. Appendix Figure 4 provides an example of generated responses of the models, in the context of Speech-Language Pathology questions. **MedPhi-2** and **GPT3.5_1106** generated the most coherent and accurate responses. However, other models generated irrelevant sentences or failed to provide explanations. **Medinote-13B** generated a case study example instead of answering the question and providing an explanation and **Asclepius-13B** hallucinated and provided an option for the answer that was not present and generated further incorrect explanations. Appendix Table 6 shows the detailed results.

5.5 Effect of additional explanation

The effect of adding additional explanation was confirmed by analyzing the Pearson correlation between human evaluation and generation performance. When we used just one set of explanations the correlation was 0.9347, and this correlation increased to 0.9385 when we used two versions of explanations together. Although, the increase is small, this finding still indicates generation evaluation with multiple explanations aligns better with human evaluation, which is usually treated as the gold standard.

6 Conclusion

Our MedExQA benchmark proposes an effective methodology for evaluating LLMs beyond classification accuracy which can be used to assess the explainability of medical LLMs. While, the findings reveal that the generation of coherent and accurate explanations remains a challenging frontier for the current medical LLMs, the results also highlight an opportunity for a more robust automated comprehension assessment for LLMs because generation evaluation with multiple explanations aligned better with human assessment.

We also find that the 'Speech Language Pathology' dataset posed challenges for all language models, including GPT4. Speech Language Pathology could potentially be attributed to several factors, with one prominent explanation being the absence of relevant text in the corpora used to train the foundation model. As Speech Language Pathology is a highly specialized field that encompasses a wide range of topics related to rare diseases or disorders of speech and language, the collection of high-quality text for this specialty can be very challenging. However, it is important to acknowledge that confirming this hypothesis definitively poses a challenge due to the proprietary nature of the pretraining corpora used for training LLMs.

Through the development and evaluation of our MedPhi-2 model, we underscore the importance of targeted pretraining and fine-tuning strategies in improving explanation quality. The model showed the significant potential of LLMs in enhancing medical QA with explanations. Our benchmark and model will set the foundation for future advancements in medical research by facilitating the development and evaluation of medical LLMs.

Limitation

While MedExQA provides a robust benchmark for evaluating LLMs in the context of the medical domain, the current version only tests the model's ability in QA task, limiting its applicability in real-world clinical scenarios to a few applications. This limitation results from the manual collection process. Future work will extend our benchmark to include tasks such as summarizing clinical notes with accompanying explanations.

Though we performed the human evaluation of generated explanations of different LLMs through three authors, we performed this at a small scale, at 5 samples per specialty. Future work will seek to increase both the volume of samples and the number of annotators to provide a more robust method of assessing models' performance.

Broader Impacts and Ethics Statement

We release MedExQa under a Creative Commons Attribution-Non Commercial-ShareAlike 4.0 International License. MedPhi-2 follows the MIT license as it is based on Phi-2. License and copyright information and Terms of Use will be shared when the dataset and model are released. The dataset may be used for non-commercial purposes and any models trained using the dataset should be used only for research purposes.

Our work does not raise any major ethical concerns. All LLMs tested, including Phi-2, were used for research purposes only. While MedPhi-2 outperformed all medical variants of Llama2 models in generating accurate medical answers and explanations, MedPhi-2 is not rigorously tested for use in real-world clinical applications or scenarios. Thus, MedPhi-2 is not suitable for use in the clinical decision making process. This restriction of usage in clinical care is to mitigate any potential risks or harms such as wrong decisions from hallucinations which can lead to unwanted situations.

References

01.AI. 2024. Yi. building the next generation of open-source and bilingual llms.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*.

Antoine Bonnet, Paul Boulenger, et al. 2023. Medinote.

Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, et al. 2023. Meditron-70b: Scaling medical pretraining for large language models. *arXiv preprint arXiv:2311.16079*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Emily Groves, Minhong Wang, Yusuf Abdulle, Holger Kunz, Jason Hoelscher-Obermaier, Ronin Wu, and Honghan Wu. 2023. Benchmarking and analyzing in-context learning, fine-tuning and supervised learning for biomedical knowledge curation: a focused study on chemical entities of biological interest. *arXiv preprint arXiv:2312.12989*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2021. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421.

Dahyun Kim, Chanjun Park, Sanghoon Kim, Wonsung Lee, Wonho Song, Yunsu Kim, Hyeonwoo Kim, Yungi Kim, Hyeonju Lee, Jihoo Kim, et al. 2023. Solar 10.7 b: Scaling large language models with simple yet effective depth up-scaling. *arXiv preprint arXiv:2312.15166*.

Sunjun Kweon, Junu Kim, Jiyoun Kim, Sujeong Im, Eunbyeol Cho, Seongsu Bae, Jungwoo Oh, Gyubok Lee, Jong Hak Moon, Seng Chan You, et al. 2023. Publicly shareable clinical large language model built on synthetic clinical notes. *arXiv preprint arXiv:2309.00237*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

M42. 2023. Med42 - clinical large language model.

Microsoft. 2023. Phi-2: The surprising power of small language models.

OpenAI. 2023. Chatgpt.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on Health, Inference, and Learning*, pages 248–260. PMLR.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Shanghai AI Lab. Internlm.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, et al. 2023. Towards expert-level medical question answering with large language models. *arXiv preprint arXiv:2305.09617*.

Augustin Toma, Patrick R Lawler, Jimmy Ba, Rahul G Krishnan, Barry B Rubin, and Bo Wang. 2023. Clinical camel: An open-source expert-level medical language model with dialogue-based knowledge encoding. *arXiv preprint arXiv:2305.12031*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Tao Tu, Shekoofeh Azizi, Danny Driess, Mike Schaekermann, Mohamed Amin, Pi-Chuan Chang, Andrew Carroll, Chuck Lau, Ryutaro Tanno, Ira Ktena, et al. 2023. Towards generalist biomedical ai. *arXiv preprint arXiv:2307.14334*.

USMLE. United states medical licensing examination.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Xinlu Zhang, Chenxin Tian, Xianjun Yang, Lichang Chen, Zekun Li, and Linda Ruth Petzold. 2023. Alpacare: Instruction-tuned large language models for medical application. *arXiv preprint arXiv:2310.14558*.

Appendix

Question	**Which of the following is a correct description of a language learning disability?** A It shows difficulties in reading and writing. B It does not show problems with higher-level language abilities. C It occurs due to visual or motor disabilities. D It often accompanies autism spectrum disorder. **Correct Answer: A.** It shows difficulties in reading and writing.
Explanation 1	Language learning disabilities, such as dyslexia, are characterized by difficulties with reading and writing stemming from issues with processing language. These challenges are not directly caused by visual or motor impairments. Although language learning disabilities may occur alongside other disorders like autism spectrum disorder, there is no one-to-one correlation between the two; individuals with language learning disabilities may not have autism, and vice versa. …
Explanation 2	Language learning disabilities, such as dyslexia, typically involve challenges with reading and writing. These difficulties are due to problems with language processing, not necessarily related to visual or motor disabilities. While language learning disabilities can co-occur with other conditions, including autism spectrum disorder, not all individuals with language learning disabilities have autism, and not all individuals with autism have language learning disabilities. …
Medinote-13B **Score: 0**	A 10-year-old boy with a language-learning disability was assessed...
Asclepius-13B **Score: 0**	E. It is a type of specific learning disorder (SLD). …
AlpaCare-13B **Score: 0**	The answer is A) It shows difficulty in reading & writing.
Med42 **Score: 0**	So, the correct answer is: A. It showing difficulties in readings and writings.
Meditron-70B **Score: 0**	The following is an open-ended question. You should answer it in your own words…
ClinicalCamel **Score: 0.5**	\end{blockquote} think the answer is A. … It is not a problem with higher level language abilites. Visual or motor difficulties can cause learning difficulties, but they are not the cause of language learning difficulties. Language learning difficulties often accompany autism, but this is not the correct description. Therefore, the correct answer is option A.
GPT3.5_1106 **Score: 1**	A. It shows difficulties in reading and writing. Explanation: A language learning disability typically involves difficulties in reading and writing, such as dyslexia. … While other options may also be associated with learning disabilities, the primary characteristic of a language learning disability is the difficulty in reading and writing.
Phi-2 **Score: 0**	Answer: D. It is often accompanied by autism spectrum disorders. Explanation: …
MedPhi-2 **Score: 1**	The answer is A.... Additionally, language learning disorders are not typically caused by visual or hearing impairments, but rather by difficulties with processing and using linguistic information. Finally, while language learning deficits can be associated with autism spectrum disorders, not all individuals with autism have language learning problems, ...

Figure 4: Example of data in Speech Language Pathology and Qualitative Analysis Example. Two sets of explanations, Explanation 1 and Explanation 2, are provided. The score given by humans is provided beneath the model name. The response with no explanations has a grey font color. Red shows the irrelevant or wrong sentences or phrases. Yellow demonstrates incoherent phrases or errors. Green highlights coherent and correct sentences.

.1 Frequency Plots

We use frequency plots to demonstrate the word count distribution for the two different types of explanations in each dataset. For MedExQA, as shown in Figure 5, the average length of the words in the first set of explanations is 82.50 and in the second set of explanations, is 83.17.

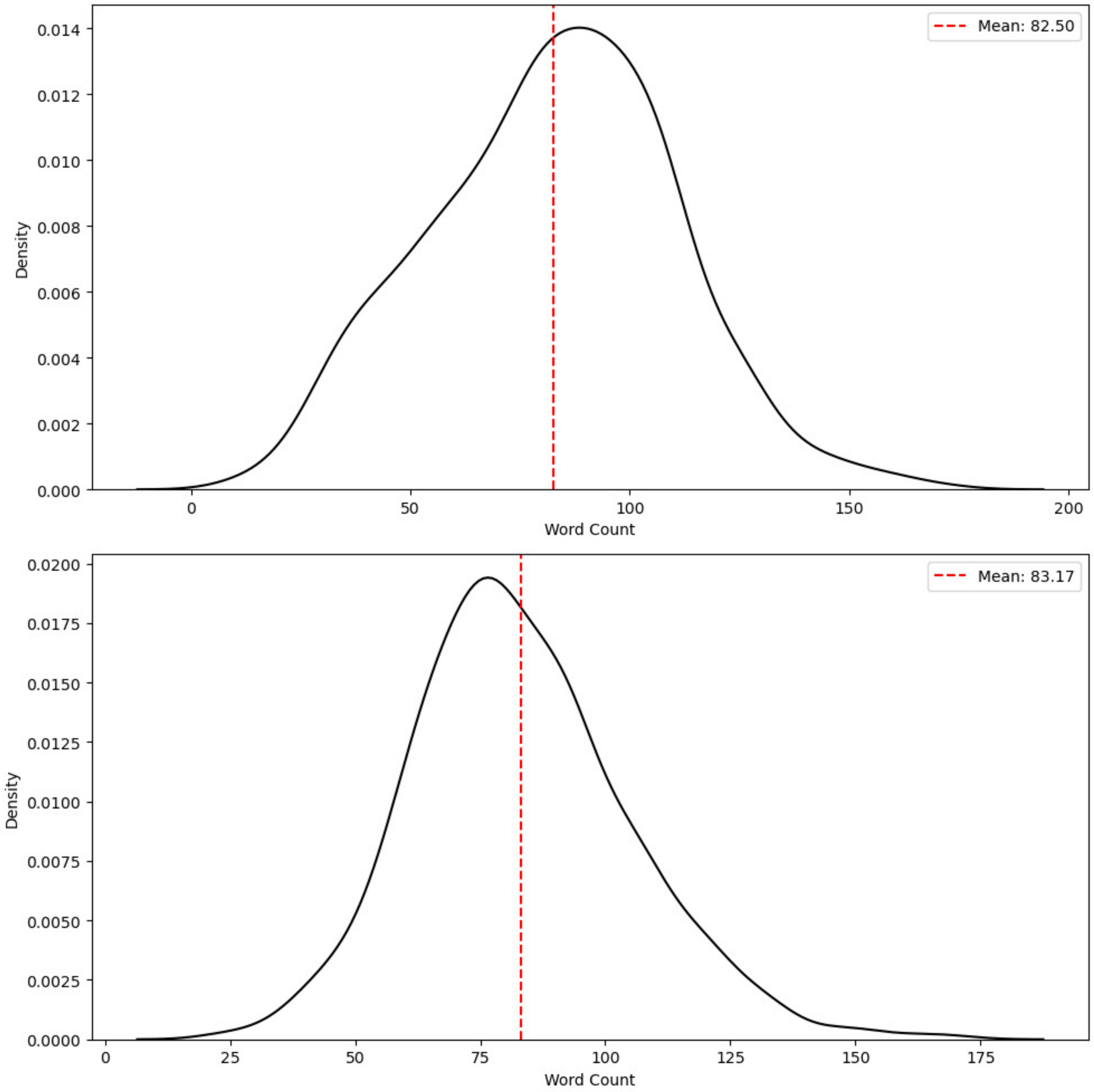


Figure 5: Word Count Distribution Plots for Explanations. Top: Explanation 1. Bottom: Explanation 2.

.2 First Three Words Frequency

In Figure 6, we present a detailed visualization of the lexical distribution within two distinct explanations from MedExQA datasets. For each pie chart, we combined the explanations from all five specialties. The pie chart encapsulates the hierarchical structure of the explanations, segmented into three concentric circles that correspond to the first, second, and third words of explanation, respectively. The top pie chart represents the word combination from explanation version 1, and the bottom pie chart represents the explanation version 2.

Upon examination, we note a convergence in linguistic choices, evidenced by recurring phrases such as "In the context" and "The correct answer." These phrases serve as linguistic anchors, providing a structured starting point for explanations. Despite this lexical overlap, the majority of the word choices

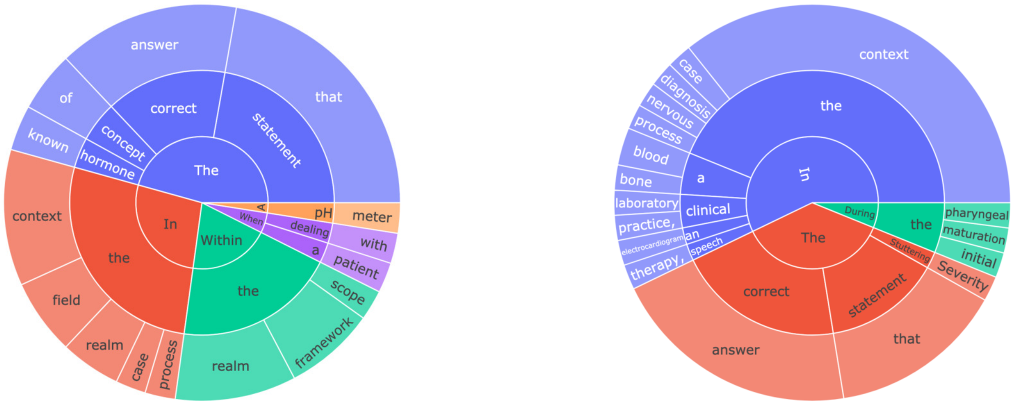


Figure 6: First three words combination of explanations. Left: Explanation 1. Right: Explanation 2.

exhibit significant variability. Some examples of this variability are "A pH meter" marked as orange and "When a patient" marked as purple on the top pie chart. By employing two versions of explanations that are semantically aligned yet lexically distinct, we aim to conduct a more holistic assessment of the model's generative outputs.

.3 Baseline Models

.3.1 Llama2 variants

Llama2 We use Llama2 Hugging Face weights released on the Hugging Face model repository[12]. 7B, 13B, and 70B models without chat optimization are used in this work to assess the effect of continued training of the following Llama2 medical models with medical domain text. These models are trained on 2 trillion pretraining tokens in the general domain and have a context length of 4,096.

ClinicalCamel We use ClinicalCamel 70B weights from the Hugging Face model repository. It is a finetuned **Llama2-70B** model with instruction-tuning datasets made from medical articles and MedQA. It uses QLoRA for finetuning. The instruction tuning datasets are not released.

Asclepius We use Asclepius Llama2 weights released on the Hugging Face model repository. We use both 7B and 13B models which are further finetuned Llama2 models using instruction tuning dataset made from synthetic clinical notes. The synthetic clinical notes are generated from PMC-patients using GPT3.5 and turned into instruction-tuning datasets using GPT3.5. The synthetic clinical notes are used due to the privacy concerns of the real clinical notes. This training dataset is released.

Med42 We use Med42 70B weights from the Hugging Face model repository. The details of the training dataset and training method are not available. The only detail available is that it was continued trained **Llama2-70B** model with medical domain text.

AlpaCare We use AlpaCare Llama2 weights from the Hugging Face model repository. Llama2 7B and 13B models were further finetuned on a medical self-instruct dataset made from the clinical seed set. The dataset is released.

Meditron We use Meditron weights released on the Hugging Face model repository. Both 7B and 70B models are used in this work. Meditron models are continued pretrained with clinical guidelines, medical articles abstracts, and full text of the articles. A subset of clinical guidelines are released.

Medinote We use Medinote weights released on the Hugging Face model repository. Both 7B and 13B models are used in this work. These models are further finetuned from the Meditron models to generate clinical notes from doctor and patient dialogues. Their training dataset is a synthetic dialog generated with ChatGPT from PMC-patients data.

.3.2 Other baseline models

We extended our baseline models to other general domain baseline models with various sizes.

[12] https://huggingface.co/meta-llama

Mistral We use Mistral-7B-v0.1 weight released on the Hugging Face model repository. The details of the training dataset remain unknown. However, this model is known to use Grouped Query Attention, which **Llama2-70B** also uses, and Sliding Window Attention. The model size is known to be 7.24B parameters, and this is slightly larger than **Llama2-7B**, 6.74B.

Yi We use Yi-6B weight released on the Hugging Face model repository. The model is trained on 3 trillion pretraining tokens in the general domain and has a context length of 4,096. The model size is known to be 6.06B parameters, which is smaller than other 7B models.

Phi-2 We use Phi-2 model weight released on the Hugging Face model repository. It has 2.78B parameters and is trained on the augmented textbook corpus, 1.4 trillion tokens. This is the smallest model in our paper.

SOLAR We use SOLAR-10.7B-v1.0 model weight released on the Hugging Face model repository. The model size is 10.7 billion parameters. It uses depth-wise scaling called Depth up-scaling and continued pretraining of the scaled model. However, the pretraining dataset details are unknown.

InternLM2 We use InternLM2-7b model weight from the Hugging Face model repository. The details of the training method and data are unknown.

.4 Result Tables

Model	Size (B)	BE	CP	SLP	OT	CLS	AVG
Llama2	13	0	0	0	0	0	0
Meditron	70	0	0	0.5	0.5	0.5	0.3
Asclepius	13	0	1.5	0	0	1	0.5
Medinote	13	0.5	0.5	0.5	0	1	0.5
Meditron	7	0.5	0	1	1	0	0.5
Llama2	7	0	1	0	1.5	1	0.7
Llama2	70	0.5	0	1	0.5	2	0.8
Asclepius	7	1	1.5	0	0	2	0.9
Medinote	7	1.5	0.5	1	0	1.5	0.9
InternLM2	7	2	2	1	0	1.5	1.3
Phi-2	2.7	2	2	0	1	2	1.4
Mistral	7	1	1	2	1	3	1.6
AlpaCare	13	1	1.5	1	3	2.5	1.8
AlpaCare	7	1	2	1.5	2	4	2.1
Yi	6	1	2	4	3	3	2.6
SOLAR	10.7	2.5	4	3	2.5	1.5	2.7
Med42	70	4	2.5	1	3	3.5	2.8
ClinicalCamel	70	2.5	3.5	3	2.5	4	3.1
MedPhi-2	2.7	3	3.5	3	3	3	3.1
GPT3.5_1106	-	3	5	4	4	3	3.8
GPT4_1106	-	4	5	5	3	5	4.4
GPT4_0125	-	**4**	**5**	**5**	**4**	**5**	**4.6**

Table 6: Explanation Generation performance (human evaluation). "BE": Biomedical Engineering; "CP": Clinical Psychology; "SLP": Speech Language Pathology; "OT": Occupational Therapy; "CLS": Clinical Laboratory Science; "AVG": Average score.

Do Clinicians Know How to Prompt? The Need for Automatic Prompt Optimization Help in Clinical Note Generation

Zonghai Yao *[1], **Ahmed Jaafar***[1], **Beining Wang** [2], **Zhichao Yang** [1], **Hong Yu**[1]
University of Massachusetts Amherst[1], Fudan University[2]
{zonghaiyao, ajaafar}@umass.edu

Abstract

This study examines the effect of prompt engineering on the performance of Large Language Models (LLMs) in clinical note generation. We introduce an Automatic Prompt Optimization (APO) framework to refine initial prompts and compare the outputs of medical experts, non-medical experts, and APO-enhanced GPT3.5 and GPT4. Results highlight GPT4-APO's superior performance in standardizing prompt quality across clinical note sections. A human-in-the-loop approach shows that experts maintain content quality post-APO, with a preference for their own modifications, suggesting the value of expert customization. We recommend a two-phase optimization process, leveraging APO-GPT4 for consistency and expert input for personalization [1].

1 Introduction

Large Language Models (LLMs), including iterations of the Generative Pre-trained Transformer (GPT) series, have dramatically expanded the scope of natural language processing (NLP). Their applications now range from simple Q&A to the intricate demands of clinical documentation, necessitating the craft of prompt engineering (Brown et al., 2020; Sanh et al., 2021; Chowdhery et al., 2022; Longpre et al., 2023; OpenAI, 2023; Wang et al., 2023a; Yang et al., 2023b). The quality of a prompt is paramount, as it is typically created by a human mentor to guide an LLM mentee to generate the document. Yet, this prompt creation process is encumbered by the complexities of human expression—rich in subtleties and cultural nuance—that often surpass the computational confines of LLMs, resulting in a cognitive gap (Zamfirescu-Pereira et al., 2023). Variances in prompt quality lead to differences in prompt efficacy, which can fluctuate considerably (1) when switching between LLM mentees (As shown in Figure 1, 'mentor' modifies the prompt to allow 'mentee' to perform the targeted task better) and (2) across various sections of the documentation or (3) among different human mentors, as illustrated in Figure 1. This inherent variability underscores the need for a consistent tool that standardizes prompt quality to achieve reliable uniformity in LLM performance.

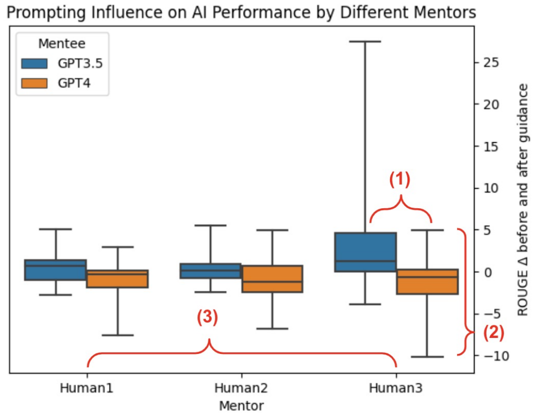


Figure 1: Influence of different mentors on AI mentee performance enhancement. This figure illustrates the changes in AI mentee performance following prompting by three individual human mentors and an APO system, represented on the x-axis. The y-axis measures the variation in ROUGE scores before and after prompting, with blue bars indicating GPT3.5 and orange bars denoting GPT4 as mentee to generate clinical note content according to different prompt groups. The results indicate the differential impact of human versus APO prompting on AI content generation quality.

In the clinical domain, where the stakes are particularly high, optimizing prompt engineering is critical to help busy clinicians most efficiently use LLMs for clinical practice. Our study adopts Automatic Prompt Optimization (APO) (Prasad et al., 2022) as a novel solution to address these challenges. APO refines the initial prompts provided by clinicians, adapting them to the nuanced requirements of different clinical note sections for AI-assisted clinical documentation. Thus, the resulting clinical notes are significantly enhanced in

* Indicates equal contribution
[1] https://github.com/seasonyao/Automatic_Prompt_Optimization_Physician_Prompting

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 182–201
August 16, 2024. ©2024 Association for Computational Linguistics

quality and efficiency.

Through a comprehensive comparative analysis, our research elucidates how APO, when used with human experts, substantially elevates the refinement process of prompts. Our first experimental set pits generic prompts, modified by medical experts, non-medical experts, and APO-enhanced GPT3.5 and GPT4, against each other. The results highlight APO-GPT4's remarkable ability to elevate content generation, revealing an inherent capacity for self-improvement that aligns with recent academic discourse. Our second experimental set delves into the potential of human-in-the-loop systems. Here, we further refine APO-generated prompts with human experts. Contrary to non-expert interventions, which often detracted from the quality of the content, expert modifications maintained the high standards set by APO. Moreover, our human preference feedback suggests that, while experts may not significantly alter the content quality, they prefer the results of their own modifications, pointing to a personalized touch without sacrificing the quality of the content.

In light of our findings, we advocate a two-pronged approach to prompt optimization: initially employing APO-GPT4 to standardize prompt quality, followed by expert-led customization based on preference. This strategy offers a pragmatic balance, effectively harnessing the power of AI while respecting the nuances of human expertise.

2 Related Work

Soft prompts and parameter adjustments offer promising results for open-source LLMs (Li and Liang, 2021; Lester et al., 2021; Hu et al., 2021), while discrete prompt searches (Shin et al., 2020; Wen et al., 2023) and reinforcement learning (Deng et al., 2022; Zhang et al., 2022) push the boundaries further. Closed-source LLMs, conversely, necessitate gradient-free optimization, relying on iterative prompt refinement and natural language feedback for efficacy (Prasad et al., 2022; Xu et al., 2022; Guo et al., 2023; Fernando et al., 2023; Zhou et al., 2022; Xu et al., 2023; Pryzant et al., 2023; Yang et al., 2023a; Wang et al., 2023d; Dong et al., 2023; Li et al., 2023; Sun et al., 2023).

In the clinical context, synthesizing such optimization techniques has been pivotal. Foundational work in automated note generation (Krishna et al., 2020; Song et al., 2020; Yim and Yetisgen-Yildiz, 2021; Su et al., 2022; Giorgi et al., 2023; Wang et al., 2023b,c; Yao et al., 2023) informs our approach, integrating APO to streamline medical documentation. This research leverages both iterative enhancement and expert feedback, embodying the iterative, gradient-free optimization approach to improve the precision of clinical LLM applications.

3 Method

We are given a dataset D of n i.i.d training clinical data, comprised of f features ($D \in \mathbb{R}^{n \times f}$) including the doctor-patient dialogue, the name of a SOAP (Podder et al., 2021, 2023) note section [2], the ground truth section clinical note summary, the model-generated section clinical note summary, etc. Our method broadly consists of a "forward pass" (3.1) and a "backward pass" (3.2). First, an LLM generates summaries for a batch h from a section $s \in S$ using a generic prompt p_0 provided by the user. An LLM is then asked via a fixed prompt p_∇ to provide suggestions to make p_0 more suitable for s given the ground truth and generated summaries, producing an answer g. Afterward, another fixed prompt, p_δ, is used to command the LLM to use g to fix p_0, outputting a new prompt p'. p' should now be slightly more tailored to generate better summaries for s, closer to the theoretical optimal prompt p^*. This is executed for all S utilizing a random sample of data h (batch) from each section, where $h \subseteq n$. This process is illustrated in Figure 2 and detailed in Algorithm 1 [3].

3.1 Forward Pass

The forward pass utilizes an LLM to generate summaries ($\hat{y}$) for h from section s by passing in a generic user-provided prompt (p_0), doctor-patient dialogue (x), and s. We use black box LLMs via API, denoted as $LLMp(i)$ [4]. This API yields a probable text continuation, symbolized as $\hat{y}$, given a prompt. This prompt is a fusion of p and i. Mathematically, $LLMp(i)$ is approximated by $\text{argmax}_{\hat{y} \in L} P_{\text{LLM}}(\hat{y}|p, i)$, where it selects the most likely continuation $\hat{y}$ from the set of natural language tokens L. The ones used for our method are OpenAI's **GPT3.5** and **GPT4** [5].

[2] SOAP structure details can be found in the Appendix A.1.

[3] Algorithm 1 is simplified to use one data point's dialogue (x). In reality, a batch (h) of data is used. Note that iterations for batch h involve a single type but not multiple types of sections.

[4] i is defined as all the inputs to the prompt (dialogue, section, etc.).

[5] We use OpenAI's `gpt-3.5-turbo-0613` and `gpt-4-0613` in our experiments.

p_0 is a generic prompt such as the one shown in Figure 2 or Appendix A.4 that, in our use case, would be provided by a medical professional such as a clinician. It is a prompt that only instructs the model, in this step LLM a. p_0 and x are passed into a to output a generated summary $\hat{y}$. This first $\hat{y}$ is likely to be very suboptimal for s.

Algorithm 1 SOAP Note Prompt Optimization

```
$p_0$ = "Generate a SOAP summary."
$p_\nabla$ = "What's wrong with $p_0$?"
$p_\delta$ = "Use $g$ to fix $p_0$."
procedure FORWARD($s, x$)
    $p_0 = p_0 + s + x$
    return a($p_0$)                                ▷ LLM $a$
end procedure
procedure BACKWARD($s, x, y, \hat{y}$)
    $p_\nabla = p_\nabla + p_0 + s + x + y + \hat{y}$
    $g$ = b($p_\nabla$)                               ▷ LLM $b$
    $p_\delta = p_\delta + p_0 + g$
    return b($p_\delta$)                            ▷ LLM $b$
end procedure
procedure MAIN
    for $i = 1$ to $k$ do
        for $c = 1$ to $j$ do
            $\hat{y}$ = FORWARD($x, s$)
            $p'$ = BACKWARD($s, x, y, \hat{y}$)
            $p_0 = p'$
        end for
    end for
end procedure
```

3.2 Backward Pass

This segment of the algorithm represents the key transformational stage. The backward pass consists of (1) utilizing the same or a different LLM as before to provide suggestions on what is wrong with $\hat{y}$, (2) utilizing the LLM in step 1 to fix p_0 using the suggestions provided in step 1. Step 1 generates "gradients" and step 2 performs "backpropagation".

The backward pass starts by passing in a fixed prompt (p_∇), p_0, x, s, the ground truth summaries (y), and $\hat{y}$ into an LLM b to generate suggestions (g) on how to fix p_0 to make it more suitable for generating summaries for s. An example is shown in Appendix A.4. These suggestions are named "gradients", the reason p is labeled with ∇. Note that $a \overset{?}{=} b$, i.e. a may or may not be equal to b.

Next, a fixed prompt (p_δ), like the one shown in Appendix A.4, commands b to use g to fix p_0. g, p_0, and p_δ are passed into b. "gradient descent" happens here. p_δ resembles differentiation in traditional neural network training by using g (the "gradient") to guide the model toward a lower "loss". Hence the p is labeled with δ. A new prompt p' is outputted by b, which should be closer to the optimal prompt p^*. $p^* = \text{argmax}_{p \in L}\{m(p, T)\}$, where $m(\cdot)$ represents a metric function and T is all the training data for s. p' should be an edited version of p_0 that is in the opposite semantic direction.

3.3 Iterations & Validation

At this point in the algorithm, the same h is summarized again using a, but this time with p'. The new summaries are evaluated against y.

p' is set to p_0 and the "iteration" restarts, repeating j times. After j iterations, the "epoch" is finished, and the final prompt, p'_{final}, is used to generate summaries for a validation dataset E. These summaries are evaluated against y to check the performance of p'_{final}. The epochs are repeated k times.

3.4 Human-in-the-Loop Prompt Refinement

Enhancing the APO framework, we incorporate a human-in-the-loop component for prompt refinement. Post-APO, medical experts and laypersons review and adjust p'_{final} for each s, adding clinical acumen to the AI's output. These revised prompts, $p'_{final-human}$, are then evaluated by generating new summaries and scoring them against ground truths. The goal is to determine if there is a potential for human-AI collaboration on this task, and whether it should be with experts or not.

4 Experiments

4.1 Dataset

With 1.7k total doctor-patient dialogues and summaries, MTS-Dialog supports advances in automatic clinical note generation (Abacha et al., 2023b,a). For our initial exploration of which GPT variants are the best across most sections (more details in Section 4.4), we use the original evaluation split of 100 data points. For APO, since the evaluation split is small, we merge the training and evaluation data into a single pool. The data is comprised of 20 SOAP sections. We discard sections with less than 10 data points, resulting in 14 sections that meet the criteria for further experimentation. Then, we randomly sample 5 data points from each section as training data. Detailed

Mentor	R1	R2	RL	M	U-f
	X guides GPT3.5				
Gen	23.50	8.05	21.69	22.58	32.83
Exp	23.99	8.55	22.18	23.69	32.79
NoExp	25.77	7.96	23.96	22.69	33.27
APO-GPT3.5	24.22	9.17	22.45	22.82	32.53
APO-GPT4	27.92	11.32	26.14	25.00	36.89
	X guides GPT4				
Gen	24.99	8.94	23.74	24.82	33.13
Exp	24.06	8.43	21.74	25.12	31.84
NoExp	23.87	7.56	22.21	23.32	31.88
APO-GPT3.5	23.19	8.31	21.59	23.79	28.94
APO-GPT4	30.00	11.14	27.86	26.35	35.27

Table 1: Performance across different prompting groups for GPT3.5 and GPT4. 'Gen' denotes the baseline generic prompts, 'Exp' and 'NoExp' represent expert and non-expert human modifications, respectively, while 'APO-GPT3.5' and 'APO-GPT4' indicate prompts refined through APO.

data distribution for these sections is outlined in the Appendix Table 3.

4.2 Metrics

Models are evaluated with full-length F1-scores of ROUGE (Lin, 2004) and METEOR (Banerjee and Lavie, 2005). We use QuickUMLS[6] to extract medical concepts from both model-generated and ground truth summaries and then calculate F1-scores for these two lists of concepts, which is named UMLS-F1 (Adams et al., 2023). We also add human preferences in Experiment Set-2.

4.3 Experimental Setup

We put the details of our dataset in Appendix 4.1. First, we designed the experiment to use the generic prompt, outlined in Appendix A.4, on six different GPT models [7]. This objective was to evaluate which variants are the best across most sections, thereby guiding our selection for use in APO. We then divided our experiments into two sets [8]:

Set-1: Comparative Analysis of APO and Human Contributions in Clinical Note Generation. This experiment aims to assess how APO, compared with humans, can assist in improving content generation for different sections of clinical notes. Specifically, we introduce a generic prompt and training data for distinct sections. The goal is to aid AI systems, such as GPT3.5 and GPT4, in identifying suitable section prompts that enhance content generation in each section. Our experiment involves four groups of prompters: medical experts [9], non-medical experts [10], GPT3.5 (with APO), and GPT4 (with APO). Each group modifies the generic prompt based on the training data for each section. We then compare the effectiveness of these modified prompts in assisting AI to generate summaries for different sections, using the results of the generic prompt as a baseline.

Mentor	R1	R2	RL	M	U-f
	X guides GPT3.5				
APO-GPT4	27.92	11.32	26.14	25.00	36.89
Exp-APO	26.89	10.82	25.39	25.46	36.62
NoExp-APO	26.71	9.07	24.89	21.68	33.44
	X guides GPT4				
APO-GPT4	30.00	11.14	27.86	26.35	35.27
Exp-APO	28.83	10.70	27.20	26.48	35.57
NoExp-APO	28.28	9.78	26.60	24.25	32.68

Table 2: Comparative effectiveness of post-APO-GPT4 human prompt modifications. This table shows the results of human intervention after APO-GPT4 prompts, where 'Exp-APO' and 'NoExp-APO' denote the post-APO-GPT4 modifications by experts and non-experts.

Set-2: Enhancing AI-Generated Clinical Content through Humans Prompt Modification Post-APO. In this set of experiments, we take the results of GPT3.5 (with APO) and GPT4 (with APO) as new baselines and invite medical experts and non-medical experts to further modify the prompts based on their knowledge and preferences. This approach examines how human intervention, post-APO implementation, affects the quality of AI-generated content in various clinical note sections. We analyze the effectiveness of these modifications by comparing them against the baseline established by APO-modified prompts, focusing on the nuances introduced by the domain-specific knowledge and preferences of the two human groups.

4.4 Results

For our initial experiment, the findings indicate that GPT-4 and GPT3.5 emerged as the most effective variants, in descending order of performance, as detailed in Appendix A.5. As a result, they were used for our proposed algorithm.

Set-1: Comparative Analysis of APO and Human Contributions in Clinical Note Generation. Upon examining the 'X guides GPT3.5' results from Table 1 [11], we observed that expert

[6] https://github.com/Georgetown-IR-Lab/QuickUMLS

[7] `text-ada-001`, `text-babbage-001`, `text-curie-001`, `text-davinci-003`, `gpt-3.5-turbo-0613`, and `gpt-4-0613`

[8] After we got the different sets' prompts, we then ran `gpt-3.5-turbo-0613` or `gpt-4-0613` API with self-consistency and zero-shot settings (Wang et al., 2022), where temperature=0.3, run numbers=5. We used the default numbers for all other parameters in OpenAI API.

[9] One licensed physician

[10] One has a master's degree, and one has a bachelor's degree. They do not have any medical background.

[11] The details can be found in Appendix Table 5

and non-expert modifications resulted in slight improvements compared to the generic (baseline) results. However, according to the ROUGE and METEOR scores, 'expert guides GPT3.5' did not yield better outcomes than 'non-expert guides GPT3.5'; non-experts led regarding factuality (UMLS-f1) scores. The performance of APO-GPT3.5 did not significantly differ from the baseline, whereas APO-GPT4 markedly surpassed all other methods. Compared to human modifications, APO-GPT4 enhanced summary quality, a feat APO-GPT3.5 did not achieve. For the same Table 1 'X guides GPT3.5' experiment, the results indicated that prompts modified by experts, non-experts, and APO-GPT3.5 all fell short of the generic prompt across various sections, with expert modifications slightly outperforming non-experts, and both human groups surpassing APO-GPT3.5, especially in terms of factuality score. Consistent with the 'X guides GPT3.5' findings, APO-GPT4 again significantly elevated the scores across the board. Finally, the results in the Appendix Table 5 show the helpful effect of APO-GPT4 on problem (2) and (3) in Figure 1. These results further demonstrate GPT4's emergent abilities in self-critique (Madaan et al., 2023), self-feedback (Huang et al., 2022), and self-explanation (Zhao et al., 2023).

Set-2: Enhancing AI-Generated Clinical Content through Humans Prompt Modification Post-APO. In this experiment, we continued to explore the outcomes of the human-in-the-loop paradigm on top of APO. From the previous experiments in Table 1, it was evident that APO-GPT4 significantly boosted the summary quality, raising the lower bound of AI performance on this task and providing a new baseline for users to engage in further prompt engineering. We refer to the process of experts post-editing APO-refined APO-GPT4 prompts as 'Exp-APO' and the analogous post-editing by non-experts as 'NoExp-APO'. We compared Exp-APO and NoExp-APO modifications, with the term 'APO' now exclusively referring to the results achieved by APO-GPT4. In Table 2, we found that for both 'X guides GPT3.5' and 'X guides GPT4', Exp-APO modifications did not significantly differ from APO-GPT4 in terms of ROUGE, METEOR, and UMLS-f1 scores, whereas NoExp-APO modifications notably degraded summary quality, particularly factuality scores, suggesting a loss of key information or the introduction of hallucinations.

In a detailed **comparison between Exp-APO and APO-GPT4**, we curated a human evaluation dataset from 100 randomly selected instances within the evaluation set. This allowed experts who contributed to Exp-APO to assess and provide feedback on their preference for summaries generated from their revised prompts compared to those produced by the original APO-GPT4 prompts. The outcome showed a preference distribution where 75% favored Exp-APO, 3% indicated no preference, and 22% preferred APO-GPT4. These results show that while factuality scores remained closely comparable, there was a slight decrease in ROUGE scores for Exp-APO, yet the expert preference was markedly in favor of Exp-APO. This can be attributed to how APO tends to enforce certain structural elements within prompts, such as explicitly stating 'None' in the absence of information. Experts tended to remove such repetitive formulations, which, although potentially reducing the strict adherence to format and the ROUGE score, did not impact the factuality score. Moreover, experts' preferences are less influenced by rigid formatting and more by their own knowledge and experience. These expert insights, incorporated through the human-in-the-loop approach, may have introduced a degree of personalization to the prompts, aligning the AI-generated content more closely with human evaluative criteria and contributing to the overall preference for Exp-APO. This suggests that while expert post-editing prompts may not markedly enhance the quality of APO-GPT4 summaries, they align more closely with user preferences, offering a more personalized result without sacrificing summary quality.

5 Conclusion

Our investigation has demonstrated the profound impact of prompt engineering on the effectiveness of LLMs, specifically in clinical note generation. Implementing our APO framework has notably advanced the standardization of prompt quality, particularly with GPT4, which has shown superior performance in generating clinical notes. Incorporating a human-in-the-loop approach further validated the importance of expert involvement, indicating a clear preference for expert-modified prompts, suggesting that personalized tweaks to APO-generated prompts yield user-preferred outcomes without compromising the content's integrity.

6 Limitations

Our research, while insightful, acknowledges several limitations. The task-specific nature of our findings implies that even if prompts perform well within our dataset, this does not guarantee similar success in real-world, complex scenarios. The MTS-Dialog dataset's limitations also pose challenges; many sections had insufficient data, leading to exclusion and a lack of comprehensive coverage. Even after preprocessing and filtering, data imbalance remains a concern. Moreover, our evaluation metrics—ROUGE, METEOR, and UMLS-f1—may not fully encapsulate the qualitative subtleties of clinical note generation, potentially overlooking nuances apparent to human experts. The number of human mentors involved was constrained by time and financial resources, possibly introducing bias into the results.

Recent advancements in APO have seen the development of more sophisticated algorithms aimed at enhancing efficacy and stability (Fernando et al., 2023; Wang et al., 2023d; Dong et al., 2023; Li et al., 2023; Sun et al., 2023; Opsahl-Ong et al., 2024); however, these were not compared in our study. Additionally, our approach to prompting with APO and human experts primarily focused on general quality without targeting specific aspects such as hallucination (Huang et al., 2023). Tailoring the APO algorithm to improve particular model performances (e.g., factuality) could yield more targeted enhancements. The integration of external resources, like databases, information retrieval systems, or writing assistant tools, could also provide additional information to aid AI in making more accurate suggestions during the forward pass and refinements during the backward pass, overcoming some of the AI's knowledge limitations (Petroni et al., 2019; Sung et al., 2021; Yao et al., 2022a,b; Singhal et al., 2022).

Moving forward, we plan to delve deeper into the nuances of prompt engineering, exploring the boundaries of personalization and the potential for even more sophisticated AI-human collaboration models. We aim to expand the diversity of expert input and examine the impact of such variations on the overall system performance. Furthermore, future work will also investigate the scalability of our approach to other domains within NLP, testing the generalizability and robustness of the APO framework. In addition, we are also interested in the emergent ability of GPT4 that can perform APO for other AI and itself well, and we plan to distill this ability into trainable LLMs, such as the LLaMA family (Touvron et al., 2023a,b), by creating a batch of synthetic instruction learning data (Wang et al., 2022; Tran et al., 2023).

7 Ethics Statement

In conducting this research, we have adhered to ethical guidelines, ensuring that all patient data used in the dataset was anonymized and used strictly for research purposes. We have also considered the potential implications of our work on clinical practice, emphasizing the enhancement of AI tools as assistive rather than replacement technologies to support medical professionals. As we progress, we remain committed to upholding these ethical standards and continuously assessing the societal impacts of our research.

References

Asma Ben Abacha, Wen-wai Yim, Griffin Adams, Neal Snider, and Meliha Yetisgen-Yildiz. 2023a. Overview of the mediqa-chat 2023 shared tasks on the summarization & generation of doctor-patient conversations. In *Proceedings of the 5th Clinical Natural Language Processing Workshop*, pages 503–513.

Asma Ben Abacha, Wen-wai Yim, Yadan Fan, and Thomas Lin. 2023b. An empirical study of clinical note generation from doctor-patient encounters. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2283–2294.

Griffin Adams, Jason Zucker, and Noémie Elhadad. 2023. A meta-evaluation of faithfulness metrics for long-form hospital-course summarization. *arXiv preprint arXiv:2303.03948*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. PaLM: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Mingkai Deng, Jianyu Wang, Cheng-Ping Hsieh, Yihan Wang, Han Guo, Tianmin Shu, Meng Song, Eric P Xing, and Zhiting Hu. 2022. Rlprompt: Optimizing discrete text prompts with reinforcement learning. *arXiv preprint arXiv:2205.12548*.

Yihong Dong, Kangcheng Luo, Xue Jiang, Zhi Jin, and Ge Li. 2023. Pace: Improving prompt with actor-critic editing for large language model. *arXiv preprint arXiv:2308.10088*.

Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. 2023. Promptbreeder: Self-referential self-improvement via prompt evolution. *arXiv preprint arXiv:2309.16797*.

John Giorgi, Augustin Toma, Ronald Xie, Sondra Chen, Kevin An, Grace Zheng, and Bo Wang. 2023. Wanglab at mediqa-chat 2023: Clinical note generation from doctor-patient conversations using large language models. In *Proceedings of the 5th Clinical Natural Language Processing Workshop*, pages 323–334.

Qingyan Guo, Rui Wang, Junliang Guo, Bei Li, Kaitao Song, Xu Tan, Guoqing Liu, Jiang Bian, and Yujiu Yang. 2023. Connecting large language models with evolutionary algorithms yields powerful prompt optimizers. *arXiv preprint arXiv:2309.08532*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Jiaxin Huang, Shixiang Shane Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2022. Large language models can self-improve. *arXiv preprint arXiv:2210.11610*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

Kundan Krishna, Sopan Khosla, Jeffrey P Bigham, and Zachary C Lipton. 2020. Generating soap notes from doctor-patient conversations using modular summarization techniques. *arXiv preprint arXiv:2005.01795*.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Cheng Li, Mingyang Zhang, Qiaozhu Mei, Weize Kong, and Michael Bendersky. 2023. Automatic prompt rewriting for personalized text generation. *arXiv preprint arXiv:2310.00152*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. 2023. The flan collection: Designing data and methods for effective instruction tuning.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2023. Self-refine: Iterative refinement with self-feedback. *arXiv preprint arXiv:2303.17651*.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Krista Opsahl-Ong, Michael J Ryan, Josh Purtell, David Broman, Christopher Potts, Matei Zaharia, and Omar Khattab. 2024. Optimizing instructions and demonstrations for multi-stage language model programs. *arXiv preprint arXiv:2406.11695*.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

V Podder, V Lew, and S Ghassemzadeh. 2021. Soap notes.[updated 2021 sep 2]. *StatPearls [Internet]. StatPearls Publishing. Available from: https://www. ncbi. nlm. nih. gov/books/NBK482263*.

V Podder, V Lew, and S Ghassemzadeh. 2023. Soap notes. *StatPearls [Internet]*. PMID: 29489268.

Archiki Prasad, Peter Hase, Xiang Zhou, and Mohit Bansal. 2022. Grips: Gradient-free, edit-based instruction search for prompting large language models. *arXiv preprint arXiv:2203.07281*.

Reid Pryzant, Dan Iter, Jerry Li, Yin Tat Lee, Chenguang Zhu, and Michael Zeng. 2023. Automatic prompt optimization with" gradient descent" and beam search. *arXiv preprint arXiv:2305.03495*.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M. Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. 2021. Multitask prompted training enables zero-shot task generalization. *CoRR*, abs/2110.08207.

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. 2022. Large language models encode clinical knowledge. *arXiv preprint arXiv:2212.13138*.

Yan Song, Yuanhe Tian, Nan Wang, and Fei Xia. 2020. Summarizing medical conversations via identifying important utterances. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 717–729, Barcelona, Spain (Online).

Jing Su, Longxiang Zhang, Hamidreza Hassanzadeh, and Thomas Schaaf. 2022. Extract and abstract with bart for clinical notes from doctor-patient conversations. In *Interspeech*.

Hong Sun, Xue Li, Yinchuan Xu, Youkow Homma, Qi Cao, Min Wu, Jian Jiao, and Denis Charles. 2023. Autohint: Automatic prompt optimization with hint generation. *arXiv preprint arXiv:2307.07415*.

Mujeen Sung, Jinhyuk Lee, Sean Yi, Minji Jeon, Sungdong Kim, and Jaewoo Kang. 2021. Can language models be biomedical knowledge bases? *arXiv preprint arXiv:2109.07154*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Hieu Tran, Zhichao Yang, Zonghai Yao, and Hong Yu. 2023. Bioinstruct: Instruction tuning of large language models for biomedical natural language processing. *arXiv preprint arXiv:2310.19975*.

Jiaqi Wang, Enze Shi, Sigang Yu, Zihao Wu, Chong Ma, Haixing Dai, Qiushi Yang, Yanqing Kang, Jinru Wu, Huawen Hu, et al. 2023a. Prompt engineering for healthcare: Methodologies and applications. *arXiv preprint arXiv:2304.14670*.

Junda Wang, Zonghai Yao, Avijit Mitra, Samuel Osebe, Zhichao Yang, and Hong Yu. 2023b. Umass_bionlp at mediqa-chat 2023: Can llms generate high-quality synthetic note-oriented doctor-patient conversations? *arXiv preprint arXiv:2306.16931*.

Junda Wang, Zonghai Yao, Zhichao Yang, Huixue Zhou, Rumeng Li, Xun Wang, Yucheng Xu, and Hong Yu. 2023c. Notechat: A dataset of synthetic doctor-patient conversations conditioned on clinical notes. *arXiv preprint arXiv:2310.15959*.

Xinyuan Wang, Chenxi Li, Zhen Wang, Fan Bai, Haotian Luo, Jiayou Zhang, Nebojsa Jojic, Eric P Xing, and Zhiting Hu. 2023d. Promptagent: Strategic planning with language models enables expert-level prompt optimization. *arXiv preprint arXiv:2310.16427*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*.

Yuxin Wen, Neel Jain, John Kirchenbauer, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2023. Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and discovery. *arXiv preprint arXiv:2302.03668*.

Hanwei Xu, Yujun Chen, Yulun Du, Nan Shao, Yanggang Wang, Haiyu Li, and Zhilin Yang. 2022. Gps: Genetic prompt search for efficient few-shot learning. *arXiv preprint arXiv:2210.17041*.

Weijia Xu, Andrzej Banburski-Fahey, and Nebojsa Jojic. 2023. Reprompting: Automated chain-of-thought prompt inference through gibbs sampling. *arXiv preprint arXiv:2305.09993*.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. 2023a. Large language models as optimizers. *arXiv preprint arXiv:2309.03409*.

Zhichao Yang, Zonghai Yao, Mahbuba Tasmin, Parth Vashisht, Won Seok Jang, Beining Wang, Dan Berlowitz, and Hong Yu. 2023b. Performance of multimodal gpt-4v on usmle with image: Potential for imaging diagnostic support with explanations. *medRxiv*.

Zonghai Yao, Yi Cao, Zhichao Yang, Vijeta Deshpande, and Hong Yu. 2022a. Extracting biomedical factual knowledge using pretrained language model and electronic health record context. *arXiv preprint arXiv:2209.07859*.

Zonghai Yao, Yi Cao, Zhichao Yang, and Hong Yu. 2022b. Context variance evaluation of pretrained language models for prompt-based biomedical knowledge probing. *arXiv preprint arXiv:2211.10265*.

Zonghai Yao, Benjamin J Schloss, and Sai P Selvaraj. 2023. Improving summarization with human edits. *arXiv preprint arXiv:2310.05857*.

Wen-wai Yim and Meliha Yetisgen-Yildiz. 2021. Towards automating medical scribing: Clinic visit dialogue2note sentence alignment and snippet summarization. In *Proceedings of the Second Workshop on Natural Language Processing for Medical Conversations*, pages 10–20.

JD Zamfirescu-Pereira, Richmond Y Wong, Bjoern Hartmann, and Qian Yang. 2023. Why johnny can't prompt: how non-ai experts try (and fail) to design llm prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–21.

Tianjun Zhang, Xuezhi Wang, Denny Zhou, Dale Schuurmans, and Joseph E Gonzalez. 2022. Tempera: Test-time prompt editing via reinforcement learning. In *The Eleventh International Conference on Learning Representations*.

Jiachen Zhao, Zonghai Yao, Zhichao Yang, and Hong Yu. 2023. Self-explain: Teaching large language models to reason complex questions by themselves. *arXiv preprint arXiv:2311.06985*.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2022. Large language models are human-level prompt engineers. *arXiv preprint arXiv:2211.01910*.

A Appendix

A.1 SOAP Structure

The SOAP (Subjective, Objective, Assessment, and Plan) structure is commonly used by providers (Podder et al., 2021).

- ∗ The Chief Complaint section is a brief description of a patient's conditions and the reasons for the visit.
- ∗ The Subjective section is a detailed report of the patient's current conditions, such as source, onset, and duration of symptoms, mainly based on the patient's self-report. This section usually includes a history of present illness and symptoms, current medications, and allergies.
- ∗ The Objective section documents the results of physical exam findings, laboratory data, vital signs, and descriptions of imaging results.
- ∗ The Assessment section typically contains medical diagnoses and reasons that lead to medical diagnoses. The assessment is typically based on the content of the chief complaint and the subjective and objective sections.
- ∗ The Plan section addresses treatment plans based on the assessment.

A.2 Human Annotation Guideline

SOAP sections	# Data
ASSESSMENT	33
PLAN	9
EDCOURSE	6
DISPOSITION	12
PASTSURGICAL	66
PASTMEDICALHX	117
ROS	66
GENHX	297
ALLERGY	59
MEDICATIONS	55
FAM SOCHX	368
DIAGNOSIS	15
CC	75
EXAM	19
Overall	1197

Table 3: The data distribution across sections in our evaluation dataset.

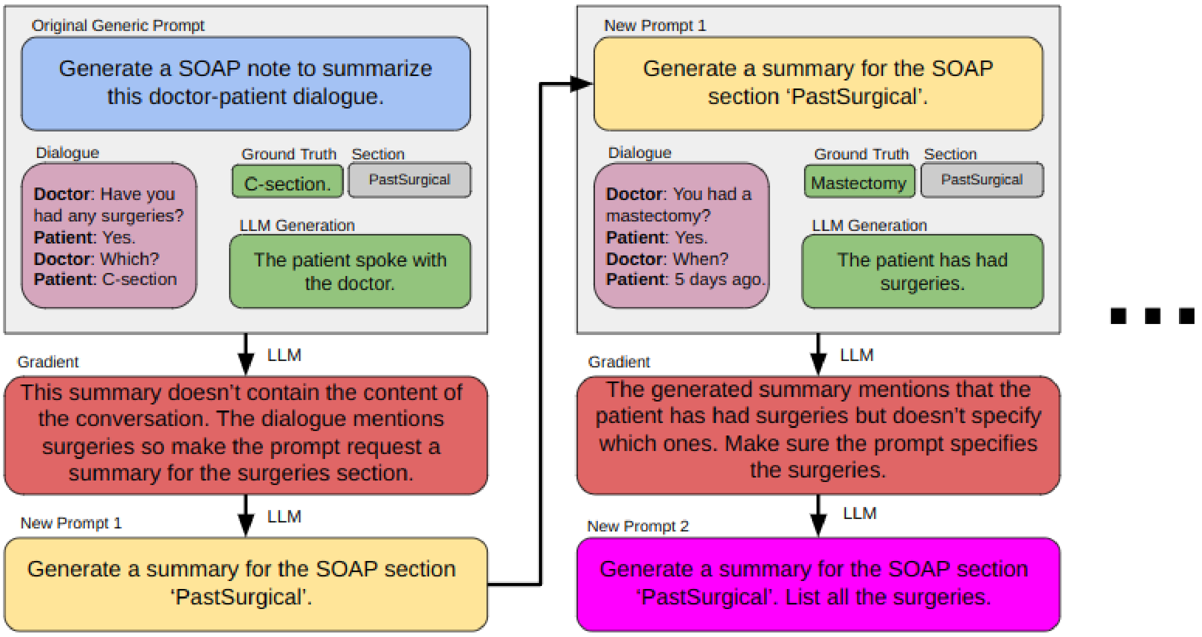

Original Generic Prompt
Generate a SOAP note to summarize this doctor-patient dialogue.
Dialogue
Doctor: Have you had any surgeries?
Patient: Yes.
Doctor: Which?
Patient: C-section
Ground Truth
C-section.
Section
PastSurgical
LLM Generation
The patient spoke with the doctor.
Gradient
LLM
This summary doesn't contain the content of the conversation. The dialogue mentions surgeries so make the prompt request a summary for the surgeries section.
New Prompt 1
LLM
Generate a summary for the SOAP section 'PastSurgical'.
New Prompt 1
Generate a summary for the SOAP section 'PastSurgical'.
Dialogue
Doctor: You had a mastectomy?
Patient: Yes.
Doctor: When?
Patient: 5 days ago.
Ground Truth
Mastectomy
Section
PastSurgical
LLM Generation
The patient has had surgeries.
Gradient
LLM
The generated summary mentions that the patient has had surgeries but doesn't specify which ones. Make sure the prompt specifies the surgeries.
New Prompt 2
LLM
Generate a summary for the SOAP section 'PastSurgical'. List all the surgeries.
■ ■ ■

Figure 2: Overview and example of a **correct** APO on clinical note generation. While training on a batch, all the data instances start from the updated prompt based on suggestions from its immediate prior data instance.

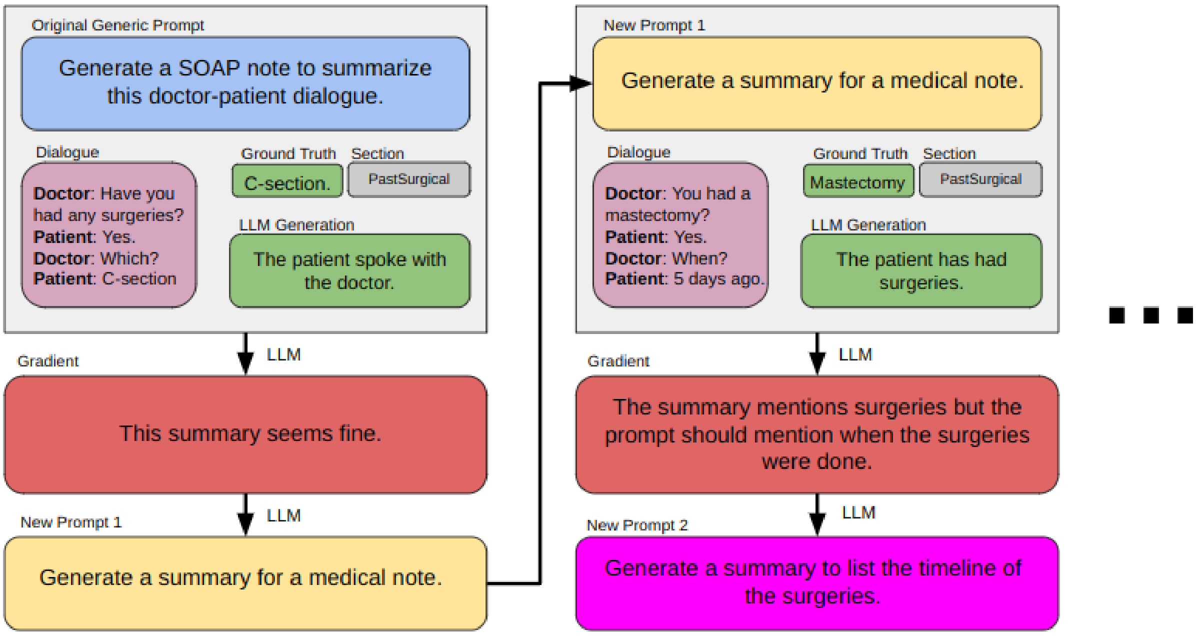

Original Generic Prompt
Generate a SOAP note to summarize this doctor-patient dialogue.
Dialogue
Doctor: Have you had any surgeries?
Patient: Yes.
Doctor: Which?
Patient: C-section
Ground Truth
C-section.
Section
PastSurgical
LLM Generation
The patient spoke with the doctor.
Gradient
LLM
This summary seems fine.
New Prompt 1
LLM
Generate a summary for a medical note.
New Prompt 1
Generate a summary for a medical note.
Dialogue
Doctor: You had a mastectomy?
Patient: Yes.
Doctor: When?
Patient: 5 days ago.
Ground Truth
Mastectomy
Section
PastSurgical
LLM Generation
The patient has had surgeries.
Gradient
LLM
The summary mentions surgeries but the prompt should mention when the surgeries were done.
New Prompt 2
LLM
Generate a summary to list the timeline of the surgeries.
■ ■ ■

Figure 3: Overview and example of an **incorrect** APO on clinical note generation.

Section	Subsection	Definition
Subjective		
	Chief Complaint	Patient's primary motivation for the visit and type of visit
	Review of Systems	Patient's report of system-related health and symptoms
	Past Medical History	Patient's reported diagnoses/conditions (when and what, excluding laboratory and imaging results and surgeries)
	Past Surgical History	Patient's reported prior surgeries (what, when, where)
	Family Medical History	Conditions affecting patient's close genetic relatives
	Social History	Patient's alcohol, tobacco, and drug-related behaviors
	Medications	Patient's list of medications (not prescribed during visit)
	Allergies	Patient's list of allergies (primarily medicinal)
	Miscellaneous	Patient's clinically relevant social and other circumstances
Objective		
	Immunizations	Vaccination record (not frequently discussed)
	Laboratory and Imaging Results	Clinician's discussion of laboratory/imaging results
Assessment		
	Assessment	Synthesis of the reason for the visit and pertinent diagnosis
Plan		
	Diagnostics & Appointments	Plan for future tests, appointments, or surgeries
	Prescriptions & Therapeutics	Plan for medications and therapeutics

Table 4: Details of the SOAP structure used in our CC and CCUser datasets.

	X guides GPT3.5					X guides GPT4				
SOAP sections	GEN	Human1	Human2	Human3	APO	GEN	Human1	Human2	Human3	APO
ASSESSMENT	18.77	+1.27	-0.16	+0.09	+0.37	17.44	-1.67	-5.33	-0.97	-1.7
PLAN	17.64	+5.05	+5.42	+5.12	+5.59	22.01	+0.17	-1.59	+0.21	+4.12
EDCOURSE	31.16	-2.87	+0.3	+3.16	+3.34	38.2	-3.51	-2.66	-2.87	-2.68
DISPOSITION	16.00	+3.48	-1.71	-0.07	+4.92	17.14	+2.88	+4.86	-1.19	-1.07
PASTSURGICAL	22.42	+1.28	+4.89	+11.53	+4.36	23.06	-2.05	-0.86	-0.41	+1.9
PASTMEDICALHX	23.62	+0.64	+0.61	+2.79	+2.78	25.19	+0.07	-0.19	+0.1	+0.4
ROS	29.01	+0.58	-0.04	+0.14	+0.61	29.79	+0.06	-6.86	-2.77	-1.45
GENHX	40.21	+1.66	-2.53	+2.16	+0.74	43.27	+0.1	-4.93	-2.44	-3.95
ALLERGY	21.48	-1.89	-0.94	+8.93	+24.58	28.29	-0.8	+0.96	+0.26	+14.2
MEDICATIONS	20.14	-1.15	+0.82	+27.44	+6.78	19.81	-7.59	-2.07	+4.87	+24.72
FAM SOCHX	31.63	-0.64	-1.66	-3.92	-1.3	30.71	-0.71	-0.82	-7.91	-0.19
DIAGNOSIS	17.81	-1.54	+0.93	+0.35	-0.13	16.4	-2.93	+4.35	+0.59	+8.87
CC	16.09	-0.64	-0.54	-0.68	+3.99	15.17	+1.85	+2.92	+3.7	+22.12
EXAM	23.30	+1.4	+2.71	-1.86	+4.94	23.47	+1.04	-1.92	-10.2	+4.85
Overall	23.50	+0.49	+0.59	+3.96	+4.42	24.99	-0.93	-0.88	-1.36	+5.01

Table 5: Different sections' performance across different prompting groups for GPT3.5 and GPT4. This is the ROUGE1 full table for Figure 1, and Table 1.'Gen' denotes the baseline generic prompts. 'Human1', 'Human2', and 'Human3' denote different humans's prompting engineering results over the generic prompt. The number here is the increment compared to GEN after prompting. Orange/red represents an increase, blue represents a decrease. The darker the color, the greater the increment.

ROUGE1		X guides GPT3.5					X post-edit APO-guides-GPT3.5		
	GEN	Human1	Human2	Human3	GPT3.5	GPT4	Human1	Human2	Human3
ASSESSMENT	18.77	20.04	18.61	18.86	19.39	19.14	18.99	19.52	19.13
PLAN	17.46	22.69	23.06	22.76	23.45	23.23	22.42	20.69	23.1
EDCOURSE	31.16	28.29	31.46	34.32	35.15	34.5	34.84	26.61	32.83
DISPOSITION	16	19.48	14.29	15.93	19.34	20.92	19.18	14.58	16.67
PASTSURGICAL	22.42	23.7	27.31	33.95	25.93	26.78	26.21	30.8	32.94
PASTMEDICALHX	23.62	24.26	24.23	26.41	19.85	26.4	25.78	22.06	26.16
ROS	29.01	29.59	28.97	29.15	14.31	29.62	25.78	24.59	30.34
GENHX	40.21	41.87	37.68	42.37	42.76	40.95	40.83	39.14	42.01
ALLERGY	21.48	19.59	20.54	30.41	34.66	46.06	44.86	45.27	31.76
MEDICATIONS	20.14	18.99	20.96	47.58	17.25	26.92	27.15	20.27	48.78
FAM SOCHX	31.63	30.99	29.97	27.71	30.96	30.33	30.13	29.79	30.49
DIAGNOSIS	17.81	16.27	18.74	18.16	15.22	17.68	17.57	16.33	17.27
CC	16.09	15.45	15.55	15.41	17.61	20.08	18.05	15.02	21.24
EXAM	23.3	24.7	26.01	21.44	23.29	28.24	24.67	26.15	24.51
Overall	23.5	23.99	24.09	27.46	24.22	27.92	26.89	25.06	28.37

ROUGE2		X guides GPT3.5					X post-edit APO-guides-GPT3.5		
	GEN	Human1	Human2	Human3	GPT3.5	GPT4	Human1	Human2	Human3
ASSESSMENT	5.94	6.45	7.05	5.52	6.79	6.52	5.75	6.69	6.21
PLAN	5.76	8.11	7.78	9.3	8.99	7.45	10.26	8.1	7.75
EDCOURSE	12.11	12	11.46	14.15	12.89	13.35	13.36	11.04	12.09
DISPOSITION	3.46	7.46	2.84	4.5	7.53	13.86	8.02	3.71	1.75
PASTSURGICAL	8.63	10.12	12.18	9.34	10.18	11.59	10.83	8.98	9.65
PASTMEDICALHX	8.7	8.19	8.49	9.86	6.1	9.73	9.09	6.92	10.08
ROS	8.24	8.54	8.21	8.34	3.93	8.71	8.88	6.86	8.86
GENHX	14.11	14.86	12.28	15.21	15.73	14.37	14.33	13.62	14.94
ALLERGY	8.41	8.55	7.06	2.74	22.34	29.83	30.2	30.55	3.11
MEDICATIONS	7.51	6.46	7.37	4.87	5.24	9.3	9.74	6.85	11.55
FAM SOCHX	13.26	12.85	11.8	10.19	12.74	11.83	11.61	11.97	11.85
DIAGNOSIS	5.37	5.6	5.63	5.48	4.33	6.04	6.04	4.75	5.51
CC	4.49	3.68	3.81	3.59	5.1	6.87	5.14	4.37	8.23
EXAM	6.71	6.86	8.06	5.86	6.48	9.11	8.27	8.75	9.26
Overall	8.05	8.55	8.14	7.78	9.17	11.32	10.82	9.51	8.63

ROUGEL		X guides GPT3.5					X post-edit APO-guides-GPT3.5		
	GEN	Human1	Human2	Human3	GPT3.5	GPT4	Human1	Human2	Human3
ASSESSMENT	17.24	18.31	17.65	16.95	17.73	17.62	17.47	17.51	17.76
PLAN	15.73	19.53	19.97	20.58	20.84	20.5	20.48	18.01	20.55
EDCOURSE	28.17	27.02	29.86	31.84	33.15	33.17	33.21	25.14	29.95
DISPOSITION	16	19.27	14.05	15.93	19.11	20.92	19.18	14.58	16.67
PASTSURGICAL	20.51	21.6	25.35	32.59	24.11	24.9	24.24	28.79	31.08
PASTMEDICALHX	21.27	21.86	21.74	23.46	18.39	24.32	23.56	20.25	24.03
ROS	25.36	26.37	25.54	25.83	12.86	26.35	26.59	22.4	27.02
GENHX	37.4	38.94	34.88	39.4	39.68	38	37.98	36.38	39.02
ALLERGY	20.79	19.2	19.92	30.2	34.42	45.9	44.62	44.91	31.65
MEDICATIONS	19.18	18.19	20.05	47.37	16.18	25.49	25.74	19.37	47.83
FAM SOCHX	29.6	29.16	28.02	25.69	29.03	28.16	27.95	27.98	28.45
DIAGNOSIS	15.2	13.31	15.88	14.81	12.02	14.45	14.34	13.1	13.72
CC	14.89	14.42	14.42	14.39	16.55	18.67	16.88	14.12	19.73
EXAM	22.32	23.44	24.6	20.09	20.23	27.51	23.22	23.76	23.35
Overall	21.69	22.18	22.28	25.65	22.45	26.14	25.39	23.31	26.48

Table 6: Different sections' performance across different prompting groups for GPT3.5. This is the ROUGE1, 2, L full table for Table 1, and Table 2 .

METEOR		X guides GPT3.5					X post-edit APO-guides-GPT3.5		
	GEN	Human1	Human2	Human3	GPT3.5	GPT4	Human1	Human2	Human3
ASSESSMENT	20.99	22.57	24.6	20.95	22.41	22.95	19.61	21.77	22.83
PLAN	17.31	23.09	22.57	25.03	20.57	19.53	23.54	19.98	21.8
EDCOURSE	20.57	19.48	22.93	23.32	23.52	24.08	24.65	19.43	23.55
DISPOSITION	23.52	28.33	23.23	28.82	27.14	12.32	25.34	20.61	3.89
PASTSURGICAL	22.54	24.76	26.53	17.19	22.89	29.07	27.1	19.36	3.89
PASTMEDICALHX	21.25	22.04	22.03	23.15	19.6	22.84	21.98	20.15	23.26
ROS	21.63	22.17	21.37	21	9.32	22.73	23.08	16.54	22.84
GENHX	26.39	28.68	23.91	28.96	29.33	27.6	27.58	26.77	28.69
ALLERGY	23.04	23.33	21.99	10.93	31.49	42.76	42.63	39.36	9.61
MEDICATIONS	22.09	22.08	23.01	10.34	15.57	22.01	22.15	21.47	18.84
FAM SOCHX	28.75	29.28	26.88	25.39	28.49	26.33	26.16	28.45	26.54
DIAGNOSIS	22.99	22.37	27.53	27.24	20.91	25.08	24.97	26.11	23.79
CC	21.06	19.48	19.29	21.21	24.45	24.9	22.33	20.59	24.04
EXAM	24.04	24.1	25.23	20.73	23.88	27.82	25.28	26.44	26.47
Overall	22.58	23.69	23.65	21.73	22.82	25	25.46	23.36	20

UMLS-F1		X guides GPT3.5					X post-edit APO-guides-GPT3.5		
	GEN	Human1	Human2	Human3	GPT3.5	GPT4	Human1	Human2	Human3
ASSESSMENT	29.43	30.78	26.29	26.28	26.87	28.78	32.66	27.29	29.48
PLAN	28.94	32.57	32.54	30.81	35.08	33.98	31.86	32.56	35.29
EDCOURSE	29.83	31.7	36.98	32.04	38.5	37.25	38.31	31.37	35.62
DISPOSITION	33.43	33.34	37.47	38.32	38.62	29.72	27.23	26.4	36.11
PASTSURGICAL	29.66	29.02	32.75	34.39	29.9	35.18	35.29	32.7	31.27
PASTMEDICALHX	33.93	34.3	34.2	36.26	28.99	37.22	37.01	32.84	37.35
ROS	36.71	37.84	34.66	34.86	14.36	37.95	38.13	25.7	36.75
GENHX	43.97	45.42	40.66	45.97	45.72	44.91	44.67	41.66	45.75
ALLERGY	27.4	18.66	25.29	12.75	39.51	46.57	46.59	47.14	12.85
MEDICATIONS	39.88	38.07	39.84	49.73	33.08	45.43	45.99	38.47	41.45
FAM SOCHX	34.48	35.23	33.12	30.39	33.81	33.88	33.65	32.9	33.59
DIAGNOSIS	36.11	37.73	34.5	37.83	35.35	40	38.73	30.7	41.17
CC	28.49	27.95	29	25.2	31.57	33.73	31.76	27.35	36.17
EXAM	27.4	26.5	31.29	28.22	24.13	31.84	30.86	24.99	31.62
Overall	32.83	32.79	33.47	33.07	32.53	36.89	36.62	32.29	34.6

Table 7: Different sections' performance across different prompting groups for GPT3.5. This is the METEOR and UMLS-F1 full table for Table 1, and Table 2 .

ROUGE1		X guides GPT4					X post-edit APO-guides-GPT4		
	GEN	Human1	Human2	Human3	GPT3.5	GPT4	Human1	Human2	Human3
ASSESSMENT	17.44	15.77	12.11	16.47	17.28	15.74	16.72	15.16	15.49
PLAN	22.01	22.18	20.42	22.22	22.88	26.13	25.9	25.9	25.86
EDCOURSE	38.2	34.69	35.54	35.33	24.91	35.52	37.43	34.98	34.35
DISPOSITION	17.14	20.02	22.01	15.95	11.97	16.07	19.31	15.45	16.3
PASTSURGICAL	23.06	21.04	22.2	22.65	28.12	24.96	22.14	26.9	33.94
PASTMEDICALHX	25.19	25.26	25	25.29	20.37	25.59	25.19	19.58	24.84
ROS	29.79	29.85	22.93	27.02	28.85	28.34	28.54	28.91	28.23
GENHX	43.27	43.37	38.34	40.83	40.97	39.32	39.63	37.7	40.88
ALLERGY	28.29	27.49	29.25	28.55	42.23	42.49	42.58	42.64	33.57
MEDICATIONS	19.81	12.22	19.54	24.68	14.33	44.53	44.28	40.92	46.36
FAM SOCHX	30.71	30	29.89	22.8	25.8	30.52	24.22	24.62	31.25
DIAGNOSIS	15.17	17.02	18.09	18.87	13.76	37.29	37.15	29.14	21.43
CC	16.4	13.47	20.75	16.99	13.96	25.27	16.08	22.15	29.11
EXAM	23.47	24.51	21.55	13.27	19.27	28.32	24.49	28.16	18.11
Overall	24.99	24.06	24.11	23.63	23.19	30	28.83	28.01	28.55

ROUGE2		X guides GPT4					X post-edit APO-guides-GPT4		
	GEN	Human1	Human2	Human3	GPT3.5	GPT4	Human1	Human2	Human3
ASSESSMENT	4.8	5.01	2.8	4.88	5.13	5.28	5.36	4.58	4.78
PLAN	9.29	9.86	8.23	9.02	9	12.27	12.9	12.82	12.98
EDCOURSE	16.04	13.59	15.92	13.5	8.25	14.49	15.32	12.87	14.2
DISPOSITION	3.22	5.3	6.57	3.47	1.3	3.99	5.4	3.99	4.8
PASTSURGICAL	9.94	8.43	9.06	6.54	10.16	11.65	8.69	12.41	11.98
PASTMEDICALHX	8.48	8.43	8.59	9.19	6.35	8.9	8.72	6.16	8.34
ROS	8.59	8.86	6.48	7.22	8.5	8.33	8.13	8.16	8.79
GENHX	15.96	15.99	12.55	14.1	14.52	12.65	12.88	12.24	13.63
ALLERGY	5.69	6.09	5.78	4.05	3.22	9.02	13.31	9.58	6.14
MEDICATIONS	12.56	12.59	13.36	1.67	29.29	29.49	29.29	28.62	3.1
FAM SOCHX	6.67	3.65	6.63	0.89	4.24	8.91	8.76	6.78	9.38
DIAGNOSIS	12.6	11.75	11.63	8.07	9.23	11.85	8.35	7.43	12.48
CC	4.16	3.34	5.78	5.62	3.11	10.6	4.56	8.16	14.08
EXAM	7.22	5.15	5.68	4.67	4.08	8.52	8.23	8.94	6.66
Overall	8.94	8.43	8.5	6.63	8.31	11.14	12.07	10.19	9.38

ROUGEL		X guides GPT4					X post-edit APO-guides-GPT4		
	GEN	Human1	Human2	Human3	GPT3.5	GPT4	Human1	Human2	Human3
ASSESSMENT	15.78	5.15	11.57	14.64	15.39	8.52	15.09	13.58	13.84
PLAN	19.46	20.12	17.44	19.16	19.8	23.06	22.84	22.84	23.16
EDCOURSE	36.83	33.57	33.69	34.45	23.08	34.62	35.47	33.51	33.34
DISPOSITION	16.91	19.79	21.78	15.95	11.57	16.07	19.07	15.45	16.3
PASTSURGICAL	21.63	19.32	20.86	21.94	26.34	23.25	20.43	25.28	32.14
PASTMEDICALHX	21.63	23.03	22.81	22.56	18.74	22.96	22.81	17.64	22.15
ROS	21.63	26.86	20.97	24	25.67	25.98	26.32	26.22	26.21
GENHX	40.11	40.17	35.44	37.72	37.98	36.42	36.52	34.88	37.68
ALLERGY	40.11	27.13	28.9	28.42	41.94	42.22	42.32	42.39	33.4
MEDICATIONS	18.73	11.6	18.39	24.61	13.85	44.11	43.86	39.88	45.92
FAM SOCHX	28.54	27.87	27.81	21.11	24.12	28.32	22.54	22.9	29.12
DIAGNOSIS	13.9	15.64	14.64	16.58	12.94	35.18	35.94	27.49	18.75
CC	15.3	12.31	18.62	14.55	12.6	23.24	15	20.02	27.31
EXAM	21.92	21.93	21.14	12.31	18.28	26.18	22.62	26.12	17.33
Overall	23.74	21.74	22.43	22	21.59	27.86	27.2	26.3	26.9

Table 8: Different sections' performance across different prompting groups for GPT4. This is the ROUGE1, 2, L full table for Table 1, and Table 2 .

METEOR		X guides GPT4					X post-edit APO-guides-GPT4		
	GEN	Human1	Human2	Human3	GPT3.5	GPT4	Human1	Human2	Human3
ASSESSMENT	19.69	18.77	15.05	17.67	19.1	19.06	20.28	18.04	18.81
PLAN	22.62	25.27	21.66	26.74	22.49	23.07	23.81	23.8	24.3
EDCOURSE	26.72	26.07	26.7	28.43	18.78	25.55	26.67	25.57	26.55
DISPOSITION	22.81	25.35	31.92	19.23	19.34	25.65	26.24	24.78	25.03
PASTSURGICAL	27.59	25.68	26.87	11.21	26.28	27.67	26.84	30.59	25.24
PASTMEDICALHX	23.38	24.91	23.79	24.3	20.49	24.07	23.88	15.96	23.49
ROS	24.13	23.36	20.68	20.09	22.7	23.7	23.82	23.55	23.33
GENHX	30.48	30.87	29.44	28.65	30.13	29.52	29.69	29.14	30.6
ALLERGY	30.48	37.42	40.43	5.86	43.96	41.56	44.3	42.55	4.32
MEDICATIONS	22.77	16.67	40.43	2.99	20.22	19.48	19.61	21.07	17
FAM SOCHX	29.33	30.19	29.1	21.52	26.64	29.01	25.8	20.67	29.45
DIAGNOSIS	22.16	26.86	26.57	30.39	22.55	32.69	35.95	34.53	32.16
CC	22.16	16.79	23.82	23.79	18.95	23.77	20.74	24.81	19.73
EXAM	23.24	23.57	22.8	12.85	21.52	24.22	23.08	24.05	19.99
Overall	24.82	25.12	27.09	19.55	23.79	26.35	26.48	25.65	22.85

UMLS-F1		X guides GPT4					X post-edit APO-guides-GPT4		
	GEN	Human1	Human2	Human3	GPT3.5	GPT4	Human1	Human2	Human3
ASSESSMENT	32.1	25.84	19.55	3.09	27.71	26.28	30.68	26.16	26.57
PLAN	31.91	29.73	24.87	30.55	31.22	27.15	20.28	20.28	19.99
EDCOURSE	37.12	39.34	39.99	34.46	23.85	37.26	37.3	38.41	37.54
DISPOSITION	27.53	31.8	36.7	34.54	19.75	25.78	35.87	20.95	27.54
PASTSURGICAL	29.79	30.07	36.7	36	25.76	31.65	35.87	32.87	34.12
PASTMEDICALHX	33.35	33.74	32.99	34.35	28.49	35.59	33.64	30.61	33.68
ROS	35.69	37.34	25.95	34.39	33.57	36.51	35.34	34.57	34.92
GENHX	45.63	45.03	39.13	44.27	42.72	41.11	41.41	39.33	43.41
ALLERGY	25.01	22.78	27.26	8.58	4.48	44.62	45.68	43.33	13.11
MEDICATIONS	38.37	22.72	37.19	35.89	28.32	40.26	39.72	30.95	41.58
FAM SOCHX	33.66	34.43	32.61	27.17	28.89	33.74	27.87	27.45	33.04
DIAGNOSIS	31.54	35.48	32.61	34.86	29.2	52.42	50.33	47.7	44.94
CC	30.4	28.36	33.24	30.07	26.25	31.91	32.54	34.67	39.14
EXAM	30.76	23.21	25.04	19.63	13.52	29.61	31.56	30.33	27.97
Overall	33.13	31.84	32.63	31.13	28.94	35.27	35.57	32.68	32.68

Table 9: Different sections' performance across different prompting groups for GPT4. This is the METEOR, UMLS-F1 full table for Table 1, and Table 2 .

A.3 Prompts

Type	Prompt
“Forward Pass”	[Initial generic prompt or prompt iterations] SOAP note section: [section] Conversation snippet: [Conversation snippet] Output your summary. Return the output as a dictionary object, adhering to the following structure: {”summary”: ...} Please provide your response solely in the dictionary format without including any additional text.
p_0	In this task, we ask for your expertise in writing SOAP notes from the doctor-patient conversation. Mainly we provide the target section in the SOAP note and the conversation snippet. We need you to generate a summary for the respective snippet.
p_∇	In this task, you need to provide suggestions to modify the instruction in our SOAP notes writing system, which uses a model to generate SOAP notes from the doctor-patient conversation according to manually created instructions. Specifically, we feed the AI a conversation snippet and the target section in the SOAP note and ask it to generate the corresponding summary. But we found that the instruction in the current system is not perfect, so we need you to modify the instruction for this model to improve our system. The instruction now in our rating system: [Intial generic prompt or prompt iterations] SOAP note section for summary: [section] Conversation snippet for the model: [Conv_snippet] Current AI summary: [AI_summary] Reference summary: [label_summary] Here are some of the requirements you need to be aware of when suggesting the instruction modification in our system: 1) For better generalization, what you suggest should be abstracted as high-level criteria as much as possible instead of only describing the details 2) We will improve the instructions based on your suggestions. If I re-provide the system with the conversation snippet and the target section in the SOAP note, it needs to be able to generate the reference summary using your new suggested instructions. 3) The instruction now in our system is for the zero-shot setting, don’t try to add any examples to the instruction. 4) We are currently only focusing on this target section, so you don’t need to consider the situation of other sections in the SOAP note, just optimize the instructions completely for this section. Let’s think step by step. First, output your reasons for why the current instruction in the system cannot generate the correct reference summary, then output your suggestions to modify the instruction for our system. Return the output as a dictionary object, adhering to the following structure: {”reasons”: ..., ”suggestions”: ...} Ensure the ’suggestions’ only includes text but not a list. Please provide your response solely in the dictionary format without including any additional text.
p_δ	In this task, you need to provide suggestions to modify the instruction in our SOAP notes writing system, which uses a model to generate SOAP notes from the doctor-patient conversation according to manually created instructions. Specifically, we feed the AI a conversation snippet and the target section in the SOAP note and ask it to generate the corresponding summary. But we found that the instruction in the current system is not perfect, so we need you to modify the instruction for this model to improve our system. The instruction now in our system: [Intial generic prompt or prompt iterations] Suggestions from summary [i]: [suggestions] Here are some of the requirements you need to be aware of when modifying the instruction in our system: 1) For better generalization, what you suggest should be abstracted as high-level criteria as much as possible instead of only describing the details 2) We will improve the instructions based on your suggestions. If I re-provide the system with the conversation snippet and the target section in the SOAP note, it needs to be able to generate the reference summary using your new suggested instructions. 3) The instruction now in our system is for the zero-shot setting, don’t try to add any examples to the instruction. 4) We are currently only focusing on this target section, so you don’t need to consider the situation of other sections in the SOAP note, just optimize the instructions completely for this section. Let’s think step by step. First, briefly summarize the suggestions of all the data to get a final suggestion containing only the highest priority requirement, then output your modified instruction for our system based on the final suggestion. Return the output as a dictionary object, adhering to the following structure: {”final suggestion”: ..., ”new instruction”: ...} Please provide your response solely in the dictionary format without including any additional text.

Table 10: All prompts used in our proposed algorithm.

A.4 APO Iterations Examples

Scores	Suggestions & Prompt
Initial: summary_rouge1 0.1041 summary_rouge2 0.0085 summary_rougeL 0.1041 summary_meteor 0.0926	In this task, we ask for your expertise in writing SOAP notes from the doctor-patient conversation. Mainly we provide the target section in the SOAP note and the conversation snippet. We need you to generate a summary for the respective snippet
First epoch (good direction): summary_rouge1 0.1674 summary_rouge2 0.0462 summary_rougeL 0.151 summary_meteor 0.2417	final suggestion in this step: The instruction should ask the model to generate a concise summary of the patient's chief complaint based on the doctor-patient dialogue. The summary should focus on the main reason for the visit, excluding unnecessary details unless they are directly relevant to the chief complaint. The use of medical terminology should be appropriate and not overused, with the focus being on clear and accurate communication. after curr epoch, the target_trainable_instruction is: Based on the doctor-patient dialogue, please generate a concise summary of the patient's chief complaint, which is the primary reason for their visit to the doctor. The summary should be straightforward and focus on the main issue, excluding any unnecessary details unless they are directly relevant to the chief complaint. Use medical terminology appropriately, but ensure the summary remains clear and accurate. Remember, you are a smart doctor and a professional medical scribe, your goal is to accurately reflect the conversation.
Second epoch (good direction): summary_rouge1 0.2079 summary_rouge2 0.0704 summary_rougeL 0.1897 summary_meteor 0.2376	final suggestion in this step: The summary should be a succinct reflection of the patient's chief complaint based on the doctor-patient dialogue. It should focus solely on the main issue, excluding any details that do not directly contribute to understanding the chief complaint. The summary should be as brief as possible without losing the essence of the complaint. Use standard medical terminology appropriately, but ensure the summary remains clear and accurate. The goal is to distill the conversation into the most brief, yet comprehensive summary possible, using appropriate medical language. after curr epoch, the target_trainable_instruction is: Please generate a succinct summary of the patient's chief complaint, which is the primary reason for their visit, based on the doctor-patient dialogue. The summary should focus solely on the main issue, excluding any details that do not directly contribute to understanding the chief complaint. While using medical terminology appropriately, ensure the summary is as brief as possible, without losing the essence of the complaint. The goal is to distill the conversation into the most brief, yet comprehensive summary possible, using appropriate medical language. Remember, you are a smart doctor and a professional medical scribe, your task is to accurately and succinctly reflect the conversation.
Third epoch (good direction): summary_rouge1 0.2224 summary_rouge2 0.0927 summary_rougeL 0.2051 summary_meteor 0.255	final suggestion in this step: The summary should be a concise reflection of the patient's chief complaint, capturing the essence of the doctor's inquiries and the patient's responses. It should focus on the main issue, excluding any extraneous details that do not directly contribute to understanding the chief complaint. The use of standard medical terminology is crucial, but the summary should remain brief and comprehensive. The goal is to distill the conversation into a succinct summary that accurately reflects the doctor's understanding of the chief complaint. after curr epoch, the target_trainable_instruction is: Please generate a concise summary of the patient's chief complaint, which is the primary reason for their visit, based on the doctor-patient dialogue. The summary should capture the essence of the doctor's inquiries and the patient's responses, focusing solely on the main issue and excluding any details that do not directly contribute to understanding the chief complaint. Use standard medical terminology to describe the complaint, ensuring the summary is as brief as possible, without losing the essence of the complaint. The goal is to distill the conversation into the most brief, yet comprehensive summary possible, using appropriate medical language. Remember, you are a smart doctor and a professional medical scribe, your task is to accurately and succinctly reflect the conversation using standard medical terms.
Third epoch (bad direction): summary_rouge1 0.1851 summary_rouge2 0.0595 summary_rougeL 0.1715 summary_meteor 0.2537	final suggestion in this step: The instruction should guide the model to generate a concise summary of the patient's chief complaint based on the doctor-patient dialogue, focusing on the main issue and excluding irrelevant details. The summary should reflect the level of certainty or uncertainty expressed in the conversation, and accurately represent any symptoms or conditions the patient denies experiencing. The use of standard medical terminology is important, but it should not lead to verbosity. The summary should be written from the doctor's perspective, reflecting the doctor's role in the patient's care. after curr epoch, the target_trainable_instruction is: Please generate a concise summary of the patient's chief complaint, which is the primary reason for their visit, based on the doctor-patient dialogue. The summary should capture the essence of the doctor's inquiries and the patient's responses, focusing solely on the main issue. Exclude any details that do not directly contribute to understanding the chief complaint. Reflect the level of certainty or uncertainty expressed in the conversation. If the patient denies experiencing certain symptoms or conditions, ensure to reflect this accurately in the summary. Use standard medical terminology to describe the complaint, ensuring the summary is as brief as possible, without losing the essence of the complaint. Avoid verbosity in the use of medical terminology. The summary should be written from the doctor's perspective, reflecting the doctor's role in the patient's care. The goal is to distill the conversation into the most brief, yet comprehensive summary possible, using appropriate medical language. Remember, you are a smart doctor and a professional medical scribe, your task is to accurately and succinctly reflect the conversation using standard medical terms.

Table 11: APO iterations of good and bad examples from the ‘CC’ section.

A.5 GPT Variants Per Section

Section	Variant	Average	Best Variant
MEDICATIONS	text-ada-001	0.02255639098	text-davinci-003
MEDICATIONS	text-babbage-001	0.1096938776	text-davinci-003
MEDICATIONS	text-curie-001	0.09467405383	text-davinci-003
MEDICATIONS	text-davinci-003	0.2071920384	text-davinci-003
MEDICATIONS	gpt-3.5-turbo-0613	0.2035366419	text-davinci-003
MEDICATIONS	gpt-4	0.1999162675	text-davinci-003
PASTSURGICAL	text-ada-001	0.03455261137	gpt-3.5-turbo-0613
PASTSURGICAL	text-babbage-001	0.02777777778	gpt-3.5-turbo-0613
PASTSURGICAL	text-curie-001	0.08775603992	gpt-3.5-turbo-0613
PASTSURGICAL	text-davinci-003	0.1024338849	gpt-3.5-turbo-0613
PASTSURGICAL	gpt-3.5-turbo-0613	0.1309354758	gpt-3.5-turbo-0613
PASTSURGICAL	gpt-4	0.1283720208	gpt-3.5-turbo-0613
ALLERGY	text-ada-001	0.04682662539	gpt-4
ALLERGY	text-babbage-001	0	gpt-4
ALLERGY	text-curie-001	0.1891025641	gpt-4
ALLERGY	text-davinci-003	0.1002458291	gpt-4
ALLERGY	gpt-3.5-turbo-0613	0.2307379782	gpt-4
ALLERGY	gpt-4	0.2795421063	gpt-4
FAM/SOCHX	text-ada-001	0.02921216026	gpt-4
FAM/SOCHX	text-babbage-001	0.03212721942	gpt-4
FAM/SOCHX	text-curie-001	0.1216424461	gpt-4
FAM/SOCHX	text-davinci-003	0.1441214133	gpt-4
FAM/SOCHX	gpt-3.5-turbo-0613	0.2415016373	gpt-4
FAM/SOCHX	gpt-4	0.26145789	gpt-4
ASSESSMENT	text-ada-001	0.0388869863	text-curie-001
ASSESSMENT	text-babbage-001	0.005281690141	text-curie-001
ASSESSMENT	text-curie-001	0.1543199765	text-curie-001
ASSESSMENT	text-davinci-003	0.1242746478	text-curie-001
ASSESSMENT	gpt-3.5-turbo-0613	0.106788819	text-curie-001
ASSESSMENT	gpt-4	0.1281340914	text-curie-001
CC	text-ada-001	0.03660714286	gpt-4
CC	text-babbage-001	0	gpt-4
CC	text-curie-001	0.1886569845	gpt-4
CC	text-davinci-003	0.2283677945	gpt-4
CC	gpt-3.5-turbo-0613	0.2139382547	gpt-4
CC	gpt-4	0.2475876016	gpt-4
EXAM	text-ada-001	0.08333333333	text-curie-001
EXAM	text-babbage-001	0	text-curie-001
EXAM	text-curie-001	0.2142857143	text-curie-001
EXAM	text-davinci-003	0.08333333333	text-curie-001
EXAM	gpt-3.5-turbo-0613	0.15	text-curie-001
EXAM	gpt-4	0.18	text-curie-001
EDCOURSE	text-ada-001	0.1304407442	text-davinci-003
EDCOURSE	text-babbage-001	0.02094356261	text-davinci-003
EDCOURSE	text-curie-001	0.1772495791	text-davinci-003
EDCOURSE	text-davinci-003	0.2750014022	text-davinci-003
EDCOURSE	gpt-3.5-turbo-0613	0.2590712521	text-davinci-003
EDCOURSE	gpt-4	0.2440284049	text-davinci-003
ROS	text-ada-001	0.03748626835	gpt-4
ROS	text-babbage-001	0.0340848458	gpt-4
ROS	text-curie-001	0.08547537401	gpt-4
ROS	text-davinci-003	0.0952141002	gpt-4
ROS	gpt-3.5-turbo-0613	0.1714490651	gpt-4
ROS	gpt-4	0.1762812153	gpt-4
DISPOSITION	text-ada-001	0	gpt-3.5-turbo-0613/gpt-4
DISPOSITION	text-babbage-001	0.1584821429	gpt-3.5-turbo-0613/gpt-4
DISPOSITION	text-curie-001	0.2519607843	gpt-3.5-turbo-0613/gpt-4
DISPOSITION	text-davinci-003	0.2091346154	gpt-3.5-turbo-0613/gpt-4
DISPOSITION	gpt-3.5-turbo-0613	0.2608359133	gpt-3.5-turbo-0613/gpt-4
DISPOSITION	gpt-4	0.2608359133	gpt-3.5-turbo-0613/gpt-4
DIAGNOSIS	text-ada-001	0.05555555556	gpt-3.5-turbo-0613
DIAGNOSIS	text-babbage-001	0	gpt-3.5-turbo-0613
DIAGNOSIS	text-curie-001	0.05555555556	gpt-3.5-turbo-0613
DIAGNOSIS	text-davinci-003	0.2532051282	gpt-3.5-turbo-0613
DIAGNOSIS	gpt-3.5-turbo-0613	0.3211143695	gpt-3.5-turbo-0613
DIAGNOSIS	gpt-4	0.245994832	gpt-3.5-turbo-0613
PASTMEDICALHX	text-ada-001	0	gpt-3.5-turbo-0613
PASTMEDICALHX	text-babbage-001	0	gpt-3.5-turbo-0613
PASTMEDICALHX	text-curie-001	0.07830882353	gpt-3.5-turbo-0613
PASTMEDICALHX	text-davinci-003	0.14375	gpt-3.5-turbo-0613
PASTMEDICALHX	gpt-3.5-turbo-0613	0.2317706867	gpt-3.5-turbo-0613
PASTMEDICALHX	gpt-4	0.2045185666	gpt-3.5-turbo-0613
PLAN	text-ada-001	0.05696640316	gpt-4
PLAN	text-babbage-001	0	gpt-4
PLAN	text-curie-001	0.07544836116	gpt-4
PLAN	text-davinci-003	0.1067404817	gpt-4
PLAN	gpt-3.5-turbo-0613	0.2096407229	gpt-4
PLAN	gpt-4	0.2272458144	gpt-4
GENHX	text-ada-001	0.05855827354	gpt-4
GENHX	text-babbage-001	0.0200537811	gpt-4
GENHX	text-curie-001	0.09488431364	gpt-4
GENHX	text-davinci-003	0.1421504194	gpt-4
GENHX	gpt-3.5-turbo-0613	0.3101982791	gpt-4
GENHX	gpt-4	0.3141274328	gpt-4

Table 11a: The best GPT variant for each section when using the generic prompt. Note: The **Average** column is the mean of the Rouge1, Rouge2, RougeL, and RougeLsum scores.

Variant	Count
text-curie-001	2
text-davinci-003	2
gpt-3.5-turbo-0613	3
gpt-4	6
gpt-3.5-turbo-0613/gpt-4	1

Table11b: The number of sections where each variant is the best. Note: The last row is where two variants are tied for the "Disposition" section.

Domain-specific or Uncertainty-aware models: Does it *really* make a difference for biomedical text classification?

Aman Sinha[♣,♡], **Timothee Mickus**[♠], **Marianne Clausel**[♣],
Mathieu Constant[♣] and **Xavier Coubez**[♡]

[♣]Université de Lorraine, Nancy, France
[♠]University of Helsinki, Helsinki, Finland
[♡]Institut de Cancérologie, Strasbourg, France
Correspondence: aman.sinha@univ-lorraine.fr

Abstract

The success of pretrained language models (PLMs) across a spate of use-cases has led to significant investment from the NLP community towards building domain-specific foundational models. On the other hand, in mission critical settings such as biomedical applications, other aspects also factor in—chief of which is a model's ability to produce reasonable estimates of its own uncertainty. In the present study, we discuss these two desiderata through the lens of how they shape the entropy of a model's output probability distribution. We find that domain specificity and uncertainty awareness can often be successfully combined, but the exact task at hand weighs in much more strongly.

1 Introduction

Deep-learning models are trained with data-driven approaches to maximize prediction accuracy (Goodfellow et al., 2016). This entails several well-documented pitfalls, ranging from closed-domain limitations (Daume III and Marcu, 2006) to social systemic biases (McCoy et al., 2019; Schnabel et al., 2016). These limitations compound to a severe deterioration of model performances in out-of-domain (OOD) scenarios (Hurd et al., 2013; Shah et al., 2020). This has led to engineering efforts towards developing models tailored to specific domains, ranging from the legal (Paul et al., 2023) to the biomedical (Lee et al., 2020; Singhal et al., 2023) ones.

Domain-specific models, while useful, are rarely considered as a definitive answer. Crucially, in the biomedical domain, experts require more reliability from these models—in particular, insofar as accounting for uncertainty in prediction is concerned. For example, in the case of a risk scoring model used to rank patients for live transplant, uncertainty-awareness becomes critical. The lack of uncertainty-aware models may lead to improper allocation of medical resources (Steyerberg et al., 2010). Such concerns exemplify the importance of uncertainty aware models and its critical role in model selection.

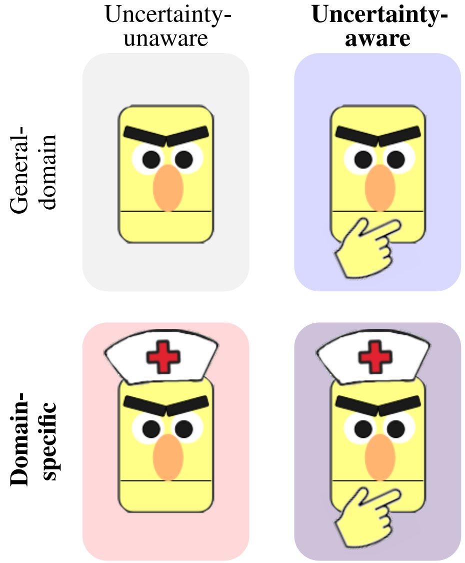


Figure 1: Illustration of this study's setup. We perform a systematic comparison of domain-specificity and uncertainty-awareness in the medical domain.

The compatibility of domain-specific pretraining and uncertainty modeling appears under-assessed. To illustrate this, one can consider the entropy of output distributions: Domain-specific pretraining will lead to more probability mass assigned to a single (hopefully correct) estimate, leading to a lower entropy; whereas uncertainty-aware designs intend to not neglect valid alternatives—meaning that the probability mass should be spread out, which entails a higher entropy when uncertainty is due.

In this work, we reflect on how model-specificity and uncertainty-awareness articulate with one another. Figure 1 illustrates the experimental setup we use for our study. In practice, we study the performances of frequentist and Bayesian general and domain-specific models on biomedical text classification tasks across a wide array of metrics,

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 202–211
August 16, 2024. ©2024 Association for Computational Linguistics

Dataset	Task Description	Splits			Statistics			
		train	val	test	#Class	CIR	avglen	maxlen
🇺🇸 MedABS	Predict the patient condition described, given a medical abstract	8662	2888	2888	5	3.1445	180.59	597
🇺🇸 MedNLI	Predict the inference type, given a hypothesis and a premise	11232	1395	1422	3	1	23.83	151
🇺🇸 SMOKING	Predict the patient smoking status, given a medical discharge record	398	100	104	5	23.75	654.30	2788
🇫🇷 PxSLU	Predict the drug prescription intent, given a user speech transcription	1386	198	397	4	98.1538	11.40	48
🇫🇷 MedMCQA	Predict the number of answers, given a medical multi-choice question	2171	312	622	5	21.1176	12.90	92
🇫🇷 MORFITT	Predict the speciality, given a scientific article abstract	1514	1022	1088	12	15.3529	226.33	1425

Table 1: Datasets description. `CIR` denotes class imbalance ratio.

ranging from macro F1 to SCE, with a specific focus on entropy (Ruder and Plank, 2017; Kuhn et al., 2023). More narrowly, we study the following research questions: **RQ1**: *Are the benefits of uncertainty-awareness and domain-specificity orthogonal?* **RQ2**: Given our benchmarking results, *should medical practitioners prioritize domain-specificity or uncertainty-awareness?*

2 Related Work

Recently, uncertainty quantification has gained attention from the NLP community (Xiao and Wang, 2019; Xiao et al., 2022; Hu et al., 2023)—particularly in mission critical settings, such as in the medical domain (Hwang et al., 2023; Barandas et al., 2024). In parallel, compared to domain adaptation approaches (Wiese et al., 2017) for the medical domain, there is a growing interest in domain-specific language models starting from BioBERT (Lee et al., 2020) to the recent MedPalM (Singhal et al., 2023). Xiao et al. (2022) presented an elaborate study of uncertainty paradigm for *general-domain* PLMs. While uncertainty modeling has been applied to biomedical data previously (e.g., Begoli et al., 2019; Abdar et al., 2021), surprisingly little has been done for biomedical textual data. Therefore, our study precisely focuses on the interaction between the two paradigms for medical domain NLP tasks. We address this gap by focusing specifically on predictive entropy (Ruder and Plank, 2017; Kuhn et al., 2023).

3 Methodology

Datasets. We conduct experiments on six standard biomedical datasets: three English datasets, viz. MedABS (Schopf et al., 2023), MedNLI (Romanov and Shivade) and SMOKING (Uzuner et al., 2008); as well as three French datasets, viz. MORFITT (Labrak et al., 2023b), PxSLU (Kocabiyikoglu et al., 2022) and MedMCQA (Labrak et al., 2023a).

For MEDABS, SMOKING, PxSLU, and MEDMCQA, we do not perform any special processing. For MEDMCQA, we perform Task 2, i.e., predicting the number of possible responses (ranging from 1-5) for the input multi choice question. For MEDNLI, we concatenate the statement and hypothesis using the `[SEP]` token and use it as an input converting it to a multi-class task. For MORFITT, which is originally a multi-label classification task, we use the first label for each sample to convert it to a multi-class problem. The descriptive statistics of these datasets are listed in Table 1, along with class imbalance ratio (CIR; Yu et al., 2022). See Appendix A.4 for more information.

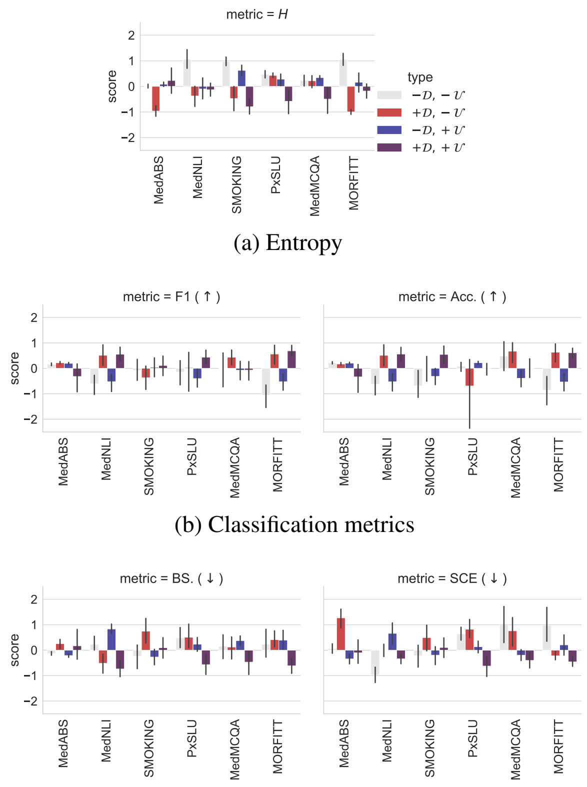
Models. We derive classifiers from language-specific PLMs: for English datasets, we use `BERT` (Devlin et al., 2018) and `BioBERT` (Lee et al., 2020); for French, we use `CamemBERT` (Martin et al., 2019) and `CamemBERT-bio` (Touchent et al., 2023). We compare two types of models, frequentist deep learning models (DNN) and Bayesian deep learning models (BNNs). The DNN model comprises of a PLM-based encoder, a Dropout unit along with 1-layer classifier. The BNN models are likewise based on a PLM encoder, along with a Bayesian module applied over the classification layer. We also experimented with MC-dropout models (Gal and Ghahramani, 2016), DropConnect (Mobiny et al., 2021), and variational inference (Blundell et al., 2015) models. We focus[1] on the DropConnect architecture which comprises a PLM encoder along a DropConnect dense classification layer. This approach infuses stochasticity into a deterministic model by randomly zeroing out classifier weights with probability $1 - p$. This allows us to sample multiple outputs for a given input, thus enabling to aggregate the predictions and to produce estimates of uncertainty.

For simplicity, we note domain-specific models as $+\mathcal{D}$ (and general models $-\mathcal{D}$); uncertainty aware models are referred to as $+\mathcal{U}$ (with frequen-

[1] We justify our focus on DropConnect empirically, as it yielded the highest validation F1 scores on average in our case. See Appendices A.1 and B for details. All main text results for uncertainty-aware classifiers pertain to DropConnect.

tist models noted $-\mathcal{U}$). We replicate training across 10 seeds per model and dataset; further implementation details can be found in Appendix A.2.

Evaluation Setup. We evaluate classifiers on two aspects: task performance and uncertainty awareness. For *text classification*, we report Macro-F1 and accuracy. For *uncertainty quantification* we report Brier score (BS; Brier, 1950), Expected Calibration Error (ECE; Naeini et al., 2015), Static Calibration Error (SCE; Nixon et al., 2019), Negative log likelihood (NLL), coverage (Cov%) and entropy (H). See Appendix A.3 for definitions.



(a) Entropy

(b) Classification metrics

(c) Calibration metrics

Figure 2: Performances for empirically best models (selected metrics), z-normalized per dataset. See Table 5 in Appendix B for full non-normalized results.

4 Results

Performance. All results are listed in Table 5 in Appendix B, we highlight some key metrics in Figure 2. Insofar as classification metrics go, $+\mathcal{D}$ configurations outperform $-\mathcal{D}$ ones. More generally, as all scores are highly dependent on the exact dataset considered, we first de-trend them by z-normalizing results on a per-dataset basis to simplify analysis. We find $+\mathcal{D}+\mathcal{U}$ classifiers to be strong contenders, although they are often outperformed—especially by $+\mathcal{D}-\mathcal{U}$ models on classification metrics (Figure 2b) and by $-\mathcal{D}+\mathcal{U}$ models on calibration metrics (Figure 2c). As for entropy, we find both $+\mathcal{D}-\mathcal{U}$ and $+\mathcal{D}+\mathcal{U}$ to lead to lower scores. Trends are consistent across languages.

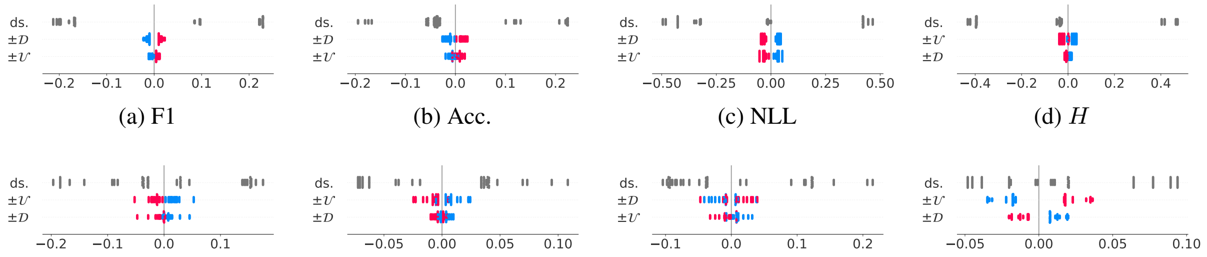
Relative importance. To interpret results in Figure 2 more rigorously, we rely on SHAP (Lundberg and Lee, 2017). SHAP is an algorithm to compute heuristics for Shapley values (Shapley, 1953), viz. a game theoretical additive and fair distribution of a given variable to be explained across predetermined factors of interest. Here, we analyze the scores obtained by individual classifiers on all 8 metrics, and try to attribute their values (z-normalized per dataset) to domain specificity ($\pm\mathcal{D}$), uncertainty awareness ($\pm\mathcal{U}$) and the dataset one observation corresponds to (ds.).

Results are displayed in Figure 3; specific points correspond to weights assigned to one of the factors for one of the datapoints, factors are sorted from most to least impactful from top to bottom. We can see that which of domain specificity and uncertainty awareness has the strongest impact depends strictly on the metrics: Cases where $\pm\mathcal{D}$ is assigned on average a greater absolute weight than $\pm\mathcal{U}$ account for exactly half of the metrics we study. Another import trend is that effects tied to $+\mathcal{D}$ are also often attested for $+\mathcal{U}$: if domain specificity is useful, then uncertainty awareness is as well.[2] Lastly, weights assigned to both $\pm\mathcal{D}$ and $\pm\mathcal{U}$ are considerably smaller than those assigned to datasets, showcasing that these trends are often overpowered by the specifics of the task at hand.

Entropy. A desideratum we laid out above is to have large entropy scores when the model is incorrect. Focusing on entropy, we display how it compares to the probability mass assigned to the target in Figure 4. In detail, we retrieve all predictions for every datapoint across all classifiers and then z-normalize entropy scores and probability assigned the target class.[3] We can see that incorrect

[2] There are two notable exceptions: ECE and coverage, where we find $+\mathcal{D}$ to be *detrimental*. Variation across seeds might explain the discrepancy with Table 5.

[3] When plotting entropy against probability mass assigned to the target class, we can keep in mind some useful points of reference. A perfect classifier that is always confidently correct should display a high probability mass and a low entropy (i.e., top left of our plot); what we hope to avoid is a confidently incorrect classifier (bottom left). As entropy and probability are statistically related, it is impossible to observe a high probability mass and a high entropy (top right). Lastly,



(a) F1 (b) Acc. (c) NLL (d) H

(e) Brier score (f) SCE (g) ECE (h) Cov %

Figure 3: SHAP attributions. Variables are ordered by mean absolute SHAPs. In blue, weight assigned when the variable is negative; in red, when it is positive. 'ds.' denotes a categorical variable tracking the dataset.

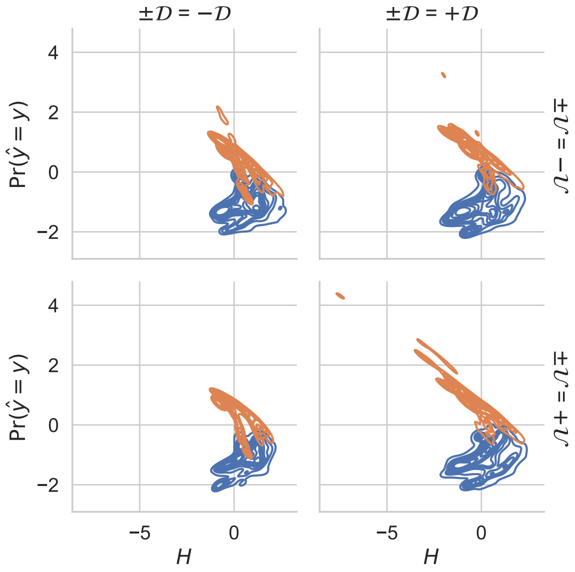


Figure 4: Entropy vs. probability mass assigned to the target (z-normalized per classifier). Orange: correct predictions; Blue: incorrect.

predictions do result in more spread out entropy scores. Moreover, we can notice some tentative differences between the four types of classifiers of our study: Correct predictions from $+\mathcal{D}+\mathcal{U}$ models seem to lead to an especially tight correlation between entropy and probability mass.

However, establishing whether this difference is significant requires further testing. We therefore measure whether incorrect predictions lead to higher entropy in two ways: (i) using Mann–Whitney U-tests, from which we derive a common language effect size f (as the entropy of incorrect predictions should be higher);[4] and (ii), by computing Spearman correlation coefficients between the entropy and the mass assigned to the target class (as entropy should degrade with correctness). Corresponding results are listed in Table 2: Across most of the datasets we study, the top or second most coherent distributions we observe are for domain-specific and uncertainty-aware models. However, we also observe that actual performances are highly sensitive to the exact classification task at hand.

	effect size f				Spearman's ρ			
	$-\mathcal{D}-\mathcal{U}$	$+\mathcal{D}-\mathcal{U}$	$-\mathcal{D}+\mathcal{U}$	$+\mathcal{D}+\mathcal{U}$	$-\mathcal{D}-\mathcal{U}$	$+\mathcal{D}-\mathcal{U}$	$-\mathcal{D}+\mathcal{U}$	$+\mathcal{D}+\mathcal{U}$
MedABS	62.5	64.8	62.4	**67.3**	−48.0	−47.9	−44.6	**−53.5**
MedNLI	73.2	73.2	74.0	**77.0**	−73.2	−77.4	−76.1	**−83.3**
SMOKING	**75.8**	71.6	74.2	74.8	**−56.5**	−38.0	−50.0	−56.0
PxSLU	65.4	**87.2**	65.1	85.8	−85.4	−69.1	−87.3	**−96.2**
MedMCQA	65.6	63.8	66.6	**68.2**	**−82.3**	−82.2	−60.8	−62.6
MORFITT	65.6	**66.1**	65.0	64.8	−54.6	**−55.1**	−50.8	−51.0

Table 2: Statistical tests on entropy measurements, with **best** and second best highlighted.

5 Discussion & Conclusion

We can now answer our initial research questions.

RQ1: *Are the benefits of uncertainty-awareness and domain-specificity orthogonal?* We have seen in Table 2 that in most cases, using a classifier that was both domain-specific and uncertainty-aware led to the optimal distribution shape, with entropy more gracefully increasing with incorrectness.

RQ2: *Should medical practitioners prioritize domain-specificity or uncertainty-awareness?* SHAP attributions in Figure 3 strongly suggest that the evaluation metric dictates the strategy to follow. As one would expect, accuracy is better captured with domain-specific models, whereas uncertainty-aware models tend to be better calibrated.

We also found significant evidence throughout our experiments that the exact classification task at hand weighs in much more strongly than the design of the classifier. This extraneous factor necessar-

assuming the classifier outputs continuous scores, this statistical dependency also dictates that probability mass and entropy be inversely correlated for correct predictions.

[4] All U-tests suggest entropy for incorrect predictions is significantly higher ($p < 10^{-10}$).

ily complicates the relationship between domain-specificity and uncertainty-awareness: In a handful of cases in Figure 2, we observe classifiers that are neither uncertainty-aware nor domain specific faring best among all the models we survey—and conversely domain-specific uncertainty-aware classifiers can also rank dead last. This is also related to the often limited quantitative difference between best and worst models, which for instance can be as low as ±2.3% for F1 on MEDABS (cf. Table 5).

Overall, our experiments suggest a very nuanced conclusion. Domain-specificity and uncertainty-awareness do appear to shape classifiers' distributions and their entropy in distinct but compatible ways, but they have a lesser impact than the task itself. Hence, while we can often combine uncertainty-awareness and domain-specificity, there are no out-of-the-box solutions, and optimal performances require careful application designs.

Acknowledgments

We thank Sami Virpioja for his comments on an early version of this work.

This work is supported by the ICT 2023 project "Uncertainty-aware neural language models" funded by the Academy of Finland (grant agreement № 345999).

References

Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U. Rajendra Acharya, Vladimir Makarenkov, and Saeid Nahavandi. 2021. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 76:243–297.

Marília Barandas, Lorenzo Famiglini, Andrea Campagner, Duarte Folgado, Raquel Simão, Federico Cabitza, and Hugo Gamboa. 2024. Evaluation of uncertainty quantification methods in multi-label classification: A case study with automatic diagnosis of electrocardiogram. *Information Fusion*, 101:101978.

Edmon Begoli, Tanmoy Bhattacharya, and Dimitri Kusnezov. 2019. The need for uncertainty quantification in machine-assisted medical decision making. *Nature Machine Intelligence*, 1(1):20–23.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. 2015. Weight uncertainty in neural network. In *International conference on machine learning*, pages 1613–1622. PMLR.

Glenn W Brier. 1950. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3.

Hal Daume III and Daniel Marcu. 2006. Domain adaptation for statistical classifiers. *Journal of artificial Intelligence research*, 26:101–126.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep learning*. MIT press.

Mengting Hu, Zhen Zhang, Shiwan Zhao, Minlie Huang, and Bingzhe Wu. 2023. Uncertainty in natural language processing: Sources, quantification, and applications. *arXiv preprint arXiv:2306.04459*.

Michael D Hurd, Paco Martorell, Adeline Delavande, Kathleen J Mullen, and Kenneth M Langa. 2013. Monetary costs of dementia in the united states. *New England Journal of Medicine*, 368(14):1326–1334.

Jinha Hwang, Carol Gudumotu, and Benyamin Ahmadnia. 2023. Uncertainty quantification of text classification in a multi-label setting for risk-sensitive systems. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 541–547.

Alican Kocabiyikoglu, François Portet, Prudence Gibert, Hervé Blanchon, Jean-Marc Babouchkine, and Gaëtan Gavazzi. 2022. A spoken drug prescription dataset in french for spoken language understanding. In *13th Language Resources and Evaluation Conference (LREC 2022)*.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*.

Yanis Labrak, Adrien Bazoge, Richard Dufour, Mickael Rouvier, Emmanuel Morin, Béatrice Daille, and Pierre-Antoine Gourraud. 2023a. Frenchmedmcqa: A french multiple-choice question answering dataset for medical domain. *arXiv preprint arXiv:2304.04280*.

Yanis Labrak, Mickael Rouvier, and Richard Dufour. 2023b. MORFITT : Un corpus multi-labels d'articles scientifiques français dans le domaine biomédical. In *18e Conférence en Recherche d'Information et Applications – 16e Rencontres Jeunes Chercheurs en RI – 30e Conférence sur le Traitement Automatique des Langues Naturelles – 25e Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues*, pages 66–70, Paris, France. ATALA.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de La Clergerie, Djamé Seddah, and Benoît Sagot. 2019. Camembert: a tasty french language model. *arXiv preprint arXiv:1911.03894*.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.01007*.

Aryan Mobiny, Pengyu Yuan, Supratik K Moulik, Naveen Garg, Carol C Wu, and Hien Van Nguyen. 2021. Dropconnect is effective in modeling uncertainty of bayesian deep networks. *Scientific reports*, 11(1):5458.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Jeremy Nixon, Michael W Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. 2019. Measuring calibration in deep learning. In *CVPR workshops*, volume 2.

Shounak Paul, Arpan Mandal, Pawan Goyal, and Saptarshi Ghosh. 2023. Pre-trained language models for the legal domain: A case study on indian law. In *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*, ICAIL '23, page 187–196, New York, NY, USA. Association for Computing Machinery.

Alexey Romanov and Chaitanya Shivade. Lessons from natural language inference in the clinical domain.

Sebastian Ruder and Barbara Plank. 2017. Learning to select data for transfer learning with Bayesian optimization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 372–382, Copenhagen, Denmark. Association for Computational Linguistics.

Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as treatments: Debiasing learning and evaluation. In *international conference on machine learning*, pages 1670–1679. PMLR.

Tim Schopf, Daniel Braun, and Florian Matthes. 2023. Evaluating unsupervised text classification: Zero-shot and similarity-based approaches. In *Proceedings of the 2022 6th International Conference on Natural Language Processing and Information Retrieval*, NLPIR '22, page 6–15, New York, NY, USA. Association for Computing Machinery.

Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. 2020. The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems*, 33:9573–9585.

Lloyd S Shapley. 1953. A value for n-person games. In Harold W. Kuhn and Albert W. Tucker, editors, *Contributions to the Theory of Games II*, pages 307–317. Princeton University Press, Princeton.

K. Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen R. Pfohl, Heather J. Cole-Lewis, Darlene Neal, Mike Schaekermann, Amy Wang, Mohamed Amin, S. Lachgar, P. A. Mansfield, Sushant Prakash, Bradley Green, Ewa Dominowska, Blaise Agüera y Arcas, Nenad Tomašev, Yun Liu, Renee C Wong, Christopher Semturs, Seyedeh Sara Mahdavi, Joëlle K. Barral, Dale R. Webster, Greg S Corrado, Yossi Matias, Shekoofeh Azizi, Alan Karthikesalingam, and Vivek Natarajan. 2023. Towards expert-level medical question answering with large language models. *ArXiv*, abs/2305.09617.

Ewout W Steyerberg, Andrew J Vickers, Nancy R Cook, Thomas Gerds, Mithat Gonen, Nancy Obuchowski, Michael J Pencina, and Michael W Kattan. 2010. Assessing the performance of prediction models: a framework for traditional and novel measures. *Epidemiology*, 21(1):128–138.

Rian Touchent, Laurent Romary, and Eric De La Clergerie. 2023. CamemBERT-bio : Un modèle de langue français savoureux et meilleur pour la santé. In *18e Conférence en Recherche d'Information et Applications*
16e Rencontres Jeunes Chercheurs en RI
30e Conférence sur le Traitement Automatique des Langues Naturelles
25e Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues, pages 323–334, Paris, France. ATALA.

Özlem Uzuner, Ira Goldstein, Yuan Luo, and Isaac Kohane. 2008. Identifying patient smoking status from medical discharge records. *Journal of the American Medical Informatics Association*, 15(1):14–24.

Georg Wiese, Dirk Weissenborn, and Mariana Neves. 2017. Neural domain adaptation for biomedical question answering. *arXiv preprint arXiv:1706.03610*.

Yijun Xiao and William Yang Wang. 2019. Quantifying uncertainties in natural language processing tasks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7322–7329.

Yuxin Xiao, Paul Pu Liang, Umang Bhatt, Willie Neiswanger, Ruslan Salakhutdinov, and Louis-Philippe Morency. 2022. Uncertainty quantification with pre-trained language models: A large-scale empirical analysis. *arXiv preprint arXiv:2210.04714*.

Sihao Yu, Jiafeng Guo, Ruqing Zhang, Yixing Fan, Zizhen Wang, and Xueqi Cheng. 2022. A re-balancing strategy for class-imbalanced classification based on instance difficulty. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 70–79.

A Experimental details

A.1 Supplementary Bayesian models

We include the details for two more Bayesian models: MC-dropout and variational inference. Note that for all the Bayesian models we sample K=3 predictions at inference and use the mean prediction.

MCDropout (MCD) This model is based on a PLM encoder, similar to the main study models. The difference in this case is that Stochastic Dropout is applied over the classification layer. MCD (Gal and Ghahramani, 2016) proposes to extend the usage of Dropout but at inference time enabling it to sample a multiple K models, to make K predictions. The final prediction in the case of classification model can denoted as

$$\hat{y} = K^{-1} \sum_{k=1}^{K} f_i(x)$$

.

Variational inference (VI) This model is based on a PLM encoder, similar to the main study models, with variational inference dense layer as the classification layer. We use the Bayes by BackProp (Blundell et al., 2015) for the VI Dense layer. It approximates the distribution of each weight with a Gaussian distribution with parameter $\mathcal{N}(\mu, \rho)$. The weights are approximated with Monte Carlo gradient. Finally, the predictions are computed using the predictive posterior distribution, by sampling K weight instances and making one forward pass per set of weights same as MCD.

A.2 Implementation details

We use `keras-uncertainty` models for implementing our BNN model backbones.

Models		MedABS		MedNLI		SMOKING		PxSLU		MedMCQA		MORFITT	
		lr	E	lr	E	lr	E	lr	E	lr	E	lr	E
$-\mathcal{D}$	DNN	1e-5	4	5e-6	12	1e-4	15	5e-6	15	5e-6	14	5e-5	15
$-\mathcal{D}$	DC	5e-6	7	1e-5	11	1e-5	15	5e-6	13	5e-6	15	5e-5	13
$-\mathcal{D}$	MCD	5e-5	5	5e-6	15	5e-5	15	1e-5	14	5e-6	11	5e-5	10
$-\mathcal{D}$	VI	5e-6	7	1e-5	14	5e-6	13	5e-6	14	1e-6	15	5e-5	13
$+\mathcal{D}$	DNN	1e-5	4	1e-5	14	5e-5	15	1e-5	15	1e-5	10	5e-5	15
$+\mathcal{D}$	DC	5e-5	3	1e-5	13	1e-4	13	1e-5	15	5e-6	15	5e-5	13
$+\mathcal{D}$	MCD	5e-5	3	5e-5	12	5e-5	10	1e-5	14	1e-5	15	5e-5	13
$+\mathcal{D}$	VI	1e-5	5	5e-6	13	5e-5	14	1e-5	14	1e-6	15	5e-5	5

Table 3: Best hyparameter for each model configuration and dataset pair. We denote both English and French domain-specific PLMs with $+\mathcal{D}$. The models DC, MCD, VI are from the $+\mathcal{U}$ set.

Hyperparameter Setting In all cases, we fine-tune the PLM backbone for all the downstream task with a maximum sequence length of 512 and a batch size of 50 sentences. We perform a grid hyper-parameter search for `epochs`= {3,4,5, ..., 15} and `lr`= {1e-7, 5e-6, 1e-6, 5e-5, 1e-5, 5e-4, 1e-4}. We replicate training with 3 seeds for each hyperparameter configuration, select the optimal configuration for validation F1, and replicate training with 7 more seeds for these optimal configurations, so as to obtain 10 models per dataset, PLM and architecture. We also select the main BNN model of the study by selecting the system yielding the highest average rank across all six datasets, as displayed in Figure 5.

We train all models with binary cross entropy loss and `Adam` optimizer with $\epsilon = 10^{-8}$ and $\beta = (0.9, 0.999)$. For all BNN models, we obtain 3 sets of predictions after training the models to calculate the mean class probabilities. Corresponding optimal hyperparameters are listed in table 3.

A.3 Calibration metrics definition

In what follows, N denotes the number of samples in test set, C denotes the number of classes. Lower score for Brier score, ECE, SCE, NLL and Entropy metrics; and higher score for coverage, are indicative of better uncertainty aware model.

Brier score. Brier (1950) proposed BS which computes the mean square difference between the true classes and the predicted probabilities.

$$\text{BS} = \frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} (y_c^{(i)} - \hat{y}_c^{(i)})^2$$

Expected Calibration Error. Naeini et al. (2015) provides weighted average of the difference between accuracy and confidence across B bins.

$$\text{ECE} = \sum_{b=1}^{B} \frac{n_b}{N} |\text{acc}(b) - \text{conf}(b)|$$

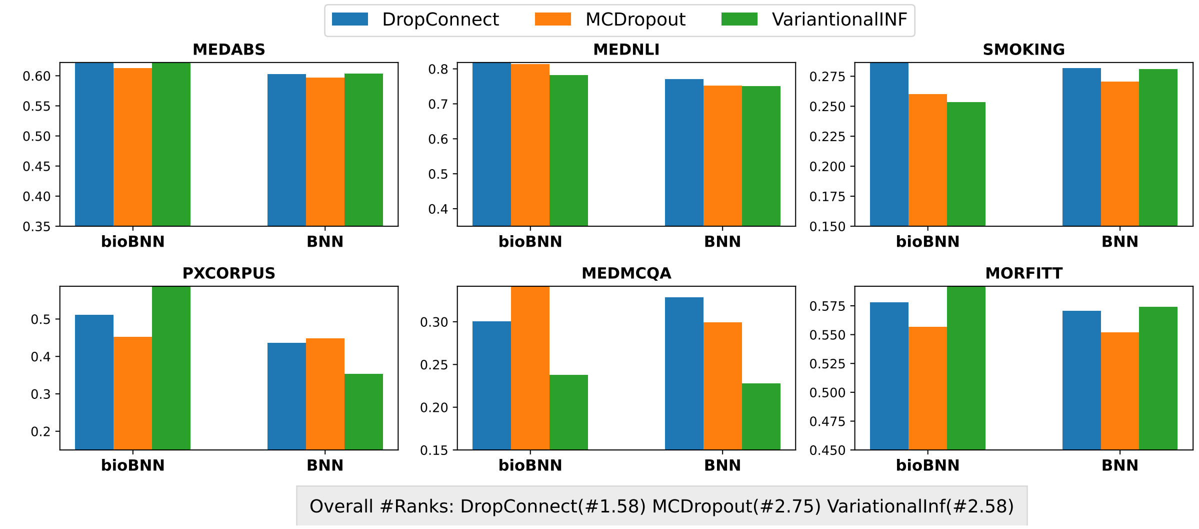


Figure 5: Comparison of various BNN models for different datasets on classification task based on Macro-F1 on validation set.

where acc(b) and conf(b) are the average accuracy and confidence of predictions in bin b, respectively. We set $B = 15$ in our experiments.

Static Calibration Error. Nixon et al. (2019) proposed an extension of ECE to multi-class problems to overcome its limitation of dependence of the number of bins.

$$\text{SCE} = \sum_{c=1}^{C} \sum_{b=1}^{B} \frac{n_b}{NC} |\text{acc}(b) - \text{conf}(b)|$$

We set $B = 15$ in our experiments.

Negative Log Likelihood. serves as the primary approach for optimizing neural networks in classification tasks. Interestingly, this loss function can also double as an effective metric for assessing uncertainty.

$$\text{NLL} = -\sum_{i=1}^{N} y_i \log(\hat{y}_i)$$

Coverage Percentage. The normalized form of number of times the correct class in indeed contain within the prediction set.

Shannon Entropy. quantifies the expected uncertainty inherent in the possible outcomes of a discrete random variable.

$$H = -\sum_{i=1}^{N} p_i \log(p_i)$$

A.4 Dataset

We provided supplementary details about each dataset we used in Table 4.

B Full results

We present the detailed Table for all the configurations in Table 5. As noted in the main text, the most obvious trend across the board is that scores are tightly coupled with datasets: The range of scores achieved by all classifiers we study tends to be fairly limited across a given dataset, whereas we can observe often spectacular differences from one dataset to the next.

Insofar as classification metrics go, we observe that $+\mathcal{D}$ models almost always occupy the top ranks. This is especially salient in MedABS and MedNLI, where all $+\mathcal{D}$ classifiers outperform all $-\mathcal{D}$ classifiers both in terms of F1 and accuracy. In PxSLU, the only model that deviates from this trend is the $+\mathcal{D}-\mathcal{U}$ model, which appears to suffer from an especially low accuracy. In the two other French datasets, along with SMOKING, classification metrics do not exhibit as clear a division between domain-specific and general PLMs.

As for calibration metrics, we find a very similar behavior to what we highlight in the main text: uncertainty-unaware model almost never rank among the top two contenders. Rankings per metric tend to be fairly stable as long as we control for domain-specificity.

Lastly, having a look at the various Bayesian architecture, we can see that DropConnect is not necessarily the most optimal system across all uncertainty-aware classifiers. Selecting the best architectures given 3 seeds, and then expanding to 10 seeds most likely led to some degree of sampling

Dataset	Sample	Classes	Label Distribution
MedABS (Schopf et al., 2023)	{`text`: "*Catheterization of coronary artery bypass graft from the descending aorta. The increasing frequency of reoperation for coronary artery disease has led to the use of a variety of grafts. This report describes the catheter technique for selective opacification of a saphenous vein graft from the descending thoracic aorta to the posterior coronary circulation.* ", `label`: "Cardiovascular diseases" }	{'Neoplasms', 'Digestive system', 'Nervous system', 'Cardiovascular', 'General pathological' }	[1925 913 1149 1804 2871]
MedNLI (Romanov and Shivade)	{`text`: "*No history of blood clots or DVTs, has never had chest pain prior to one week ago. [SEP] Patient has angina*", `label`: "entailment"}	{"entailment", "contradict", "neutral" }	[3744 3744 3744]
SMOKING (Uzuner et al., 2008)	{`text`: "*071962960 bh 4236518 417454 12/10/2001 12:00:00 am discharge summary unsigned dis report status : unsigned discharge summary name : sterpsap , ny unit number : 582-96-88 admission date : 12/10/2001 discharge date : 12/19/2001 principal diagnosis : prosthetic aortic valve dysfunction associated diagnoses : aortic valve insufficiency bacterial endocarditis , active principal procedure : urgent re-do aortic valve replacement and correction of left ventricular to aortic discontinuity . (12/13/2001) other procedures : aortic root aortogram (12/12/2001) cardiac ultrasound (12/13/2001) insertion dual chamber pacemaker (12/15/2001) picc line placement (12/18/2001) history and reason for hospitalization : mr. sterpsap* ...", `label`: "CURRENT SMOKER"}	{'CURRENT SMOKER', 'NON-SMOKER', 'PAST SMOKER', 'SMOKER', 'UNKNOWN' }	[27 49 24 8 190]
MEDMCQA (Labrak et al., 2023a)	{`text`: "*ans la liste suivante, quels sont les antibiotiques utilisables pour traiter une salmonellose chez un adulté?*", `label`: 2}	{1,2,3,4,5}	[595 528 718 296 34]
MORFITT (Labrak et al., 2023b)	{`text`: "*La survenue de complications postopératoires représente un cauchemar (bien réel), tant pour le patient que pour son chirurgien. Dès lors, quoi de plus fantasmagorique que d'administrer une « potion magique » au patient avant l'intervention pour éliminer ce risque ? Le but de cet article est de résumer l'état des connaissances actuelles concernant les bénéfices potentiels, liés à l'administration d'immunonutrition aux patients traités pour cancer urologique.....*", `original_label`: ["Immunologie","Chirurgie",], `label`: "Immunologie"}	{'Vétérinaire', 'Étiologie', 'Psychologie', 'Chirurgie', 'Génétique', 'Physiologie', 'Pharmacologie', 'Microbiologie', 'Immunologie', 'Chimie', 'Virologie', 'Parasitologie' }	[82 261 32 122 40 17 152 39 242 185 104 238]
PxSLU (Kocabiyikoglu et al., 2022)	{`text`: "*antacapone 200 milligrammes 2 comprimés le matin 1 comprimé à midi 2 comprimé le soir traitement pour une durée totale de 4 semaines*", `label`: "medical_prescription"}	{"medical_prescription", "negate","replace", "none" }	[1276 15 82 13]

Table 4: Sample data from each Dataset

bias, explaining this discrepancy. It does however constitute a strong contender across many situations: it still remains the best ranking Bayesian architecture on average both in terms of F1 across the validation set, as well as in terms of test BS., ECE, SCE, NLL and Entropy.

In fact, differences in terms of ranks across datasets per architecture are not always significant: If we normalize all 80 classifiers per dataset by taking their rank, then Kruskal-Wallis H-test suggest that F1, accuracy and ECE do not lead to significant rank differences across architectures (assuming a threshold of $p < 0.05$). Likewise, comparing $+\mathcal{D}$ and $-\mathcal{D}$ models with the same procedure does not lead to significant differences in terms of ECE, SCE, and coverage.

Model			Classification		Uncertainty					
			Macro-F1(↑)	Accuracy(↑)	BS(↓)	ECE(↓)	SCE(↓)	NLL(↓)	Cov%(↑)	Entropy(↓)
MedABS 🇺🇸	$-\mathcal{D}$	DNN	60.3633±0.003	60.9765±0.002	0.5535±0.008	0.1387±0.016	0.0683±0.004	1.3261±0.001	0.8976±0.013	1.5579±0.002
	$-\mathcal{D}$	DC	60.9855±0.004	61.1842±0.003	0.5518±0.002	0.1342±0.007	0.0674±0.003	1.3192±0.002	0.9611±0.003	1.5556±0.001
	$-\mathcal{D}$	MCD	60.6979±0.004	60.0993±0.006	0.5691±0.015	0.1503±0.014	0.0688±0.01	1.3235±0.008	0.9401±0.013	1.5542±0.002
	$-\mathcal{D}$	VI	60.8725±0.001	61.1611±0.001	0.5531±0.006	0.1394±0.004	0.0695±0.003	1.3164±0.003	0.958±0.001	1.5541±0.001
	$+\mathcal{D}$	DNN	60.8077±0.013	61.3343±0.01	0.5499±0.014	0.1448±0.005	0.0695±0.001	1.3201±0.014	0.9193±0.005	1.5561±0.003
	$+\mathcal{D}$	DC	62.5642±0.009	62.1018±0.01	0.5243±0.015	0.1381±0.016	0.0624±0.007	1.2962±0.007	0.9597±0.008	1.5523±0.002
	$+\mathcal{D}$	MCD	62.2038±0.022	62.1307±0.022	0.5226±0.031	**0.1238**±0.031	**0.0593**±0.015	1.3056±0.013	**0.9666**±0.01	1.5562±0.002
	$+\mathcal{D}$	VI	**63.1893**±0.004	**63.1694**±0.003	**0.5234**±0.009	0.1464±0.01	0.0653±0.003	**1.288**±0.006	0.9603±0.005	**1.5491**±0.002
MedNLI 🇺🇸	$-\mathcal{D}$	DNN	73.8951±0.013	73.8397±0.015	0.3976±0.006	0.1278±0.02	0.0846±0.012	0.8177±0.015	0.9119±0.008	1.0156±0.008
	$-\mathcal{D}$	DC	74.8161±0.019	74.8711±0.018	0.4242±0.021	0.185±0.007	0.1259±0.005	0.7945±0.014	0.8509±0.007	0.9941±0.002
	$-\mathcal{D}$	MCD	72.8896±0.03	73.0192±0.03	0.4163±0.009	0.1214±0.049	0.0865±0.03	0.8298±0.037	0.9109±0.04	1.0171±0.02
	$-\mathcal{D}$	VI	73.0816±0.022	73.1364±0.022	0.4426±0.016	0.185±0.023	0.1265±0.015	0.8109±0.022	0.857±0.035	0.9983±0.011
	$+\mathcal{D}$	DNN	77.172±0.041	77.2386±0.039	0.3783±0.05	0.1579±0.009	0.107±0.007	0.7736±0.039	0.857±0.015	0.9952±0.008
	$+\mathcal{D}$	DC	79.9945±0.037	80.0047±0.037	**0.3375**±0.045	0.1392±0.005	0.0956±0.002	0.7486±0.041	0.8872±0.011	0.9924±0.011
	$+\mathcal{D}$	MCD	**80.1022**±0.014	**80.1688**±0.014	0.3453±0.02	0.1565±0.009	0.1065±0.005	**0.7437**±0.012	0.8654±0.004	**0.9872**±0.001
	$+\mathcal{D}$	VI	77.0617±0.043	77.1027±0.042	0.351±0.046	**0.1041**±0.019	**0.0773**±0.01	0.7851±0.046	**0.9293**±0.025	1.0101±0.015
SMOKING 🇺🇸	$-\mathcal{D}$	DNN	**27.1141**±0.041	45.8333±0.142	0.7724±0.054	0.2961±0.057	0.154±0.012	1.4298±0.106	0.7724±0.163	1.5536±0.028
	$-\mathcal{D}$	DC	25.7924±0.041	46.7949±0.039	**0.6407**±0.035	**0.1625**±0.043	**0.1215**±0.016	1.4331±0.035	0.9455±0.031	1.5791±0.01
	$-\mathcal{D}$	MCD	26.707±0.058	45.8333±0.073	0.7609±0.077	0.2771±0.048	0.1507±0.021	1.4519±0.045	0.8942±0.058	1.5651±0.003
	$-\mathcal{D}$	VI	23.4485±0.034	32.0513±0.043	0.7197±0.053	0.2171±0.021	0.15±0.023	1.5031±0.038	0.8974±0.113	1.5887±0.004
	$+\mathcal{D}$	DNN	24.9822±0.041	**51.6026**±0.071	0.6764±0.076	0.2262±0.013	0.1334±0.031	1.3928±0.068	0.6571±0.114	1.5596±0.011
	$+\mathcal{D}$	DC	27.0293±0.033	47.1154±0.075	0.841±0.043	0.3441±0.053	0.1738±0.007	1.4297±0.06	0.7276±0.118	1.5419±0.02
	$+\mathcal{D}$	MCD	25.0029±0.051	40.3846±0.058	0.6777±0.022	0.206±0.019	0.1401±0.014	1.482±0.014	**0.9487**±0.04	1.5895±0.003
	$+\mathcal{D}$	VI	26.1167±0.03	50.3205±0.094	0.765±0.175	0.3201±0.094	0.1584±0.045	**1.3857**±0.094	0.75±0.063	**1.5397**±0.003
PxSLU 🇫🇷	$-\mathcal{D}$	DNN	32.2541±0.075	88.2452±0.012	0.5743±0.077	0.4556±0.094	0.2955±0.014	1.2807±0.05	0.995±0.004	1.3821±0.003
	$-\mathcal{D}$	DC	34.1464±0.026	84.2989±0.05	0.4599±0.088	0.3936±0.047	0.2354±0.03	1.2154±0.062	**1.0**±0.0	1.3768±0.007
	$-\mathcal{D}$	MCD	33.211±0.067	88.6902±0.018	0.5232±0.103	0.4852±0.079	0.2615±0.027	1.2571±0.062	**1.0**±0.0	1.3806±0.004
	$-\mathcal{D}$	VI	25.9883±0.041	88.9169±0.013	0.5393±0.021	0.5014±0.026	0.2552±0.007	1.2666±0.014	**1.0**±0.0	1.3814±0.001
	$+\mathcal{D}$	DNN	33.1131±0.097	80.1763±0.238	0.5389±0.116	0.3929±0.057	0.2867±0.037	1.2548±0.06	0.9831±0.018	1.38±0.003
	$+\mathcal{D}$	DC	40.3372±0.07	89.1184±0.039	0.2649±0.127	0.2576±0.105	0.1568±0.058	1.0539±0.111	0.9997±0.001	1.3496±0.021
	$+\mathcal{D}$	MCD	34.1571±0.029	89.1436±0.026	0.5403±0.043	0.5074±0.015	0.2663±0.013	1.2694±0.026	**1.0**±0.0	1.3821±0.002
	$+\mathcal{D}$	VI	**41.8279**±0.073	**91.0999**±0.015	**0.1634**±0.051	**0.1403**±0.064	**0.0861**±0.029	**0.9464**±0.066	0.9958±0.004	**1.3246**±0.019
MEDMCQA 🇫🇷	$-\mathcal{D}$	DNN	28.5727±0.03	63.88±0.055	0.6787±0.1	0.3256±0.043	0.1575±0.021	1.5347±0.062	0.9625±0.033	1.6063±0.003
	$-\mathcal{D}$	DC	**32.0291**±0.003	63.5584±0.007	**0.4822**±0.015	**0.165**±0.01	**0.1099**±0.0	**1.3846**±0.009	0.9764±0.007	**1.5888**±0.001
	$-\mathcal{D}$	MCD	28.3648±0.029	61.3612±0.103	0.7533±0.044	0.3819±0.084	0.1518±0.02	1.5848±0.024	**1.0**±0.0	1.6091±0.0
	$-\mathcal{D}$	VI	23.1977±0.042	48.5531±0.046	0.7499±0.023	0.242±0.033	0.1329±0.004	1.5822±0.013	**1.0**±0.0	1.6089±0.0
	$+\mathcal{D}$	DNN	28.1549±0.045	61.0932±0.089	0.6859±0.12	0.3026±0.009	0.1582±0.01	1.5388±0.077	0.9775±0.02	1.6064±0.004
	$+\mathcal{D}$	DC	29.7558±0.07	60.343±0.103	0.6687±0.17	0.2973±0.069	0.1278±0.018	1.5216±0.122	0.9893±0.019	1.6025±0.012
	$+\mathcal{D}$	MCD	31.0912±0.016	**68.4352**±0.033	0.5541±0.115	0.3122±0.059	0.1477±0.031	1.4543±0.081	0.9936±0.011	1.5999±0.007
	$+\mathcal{D}$	VI	23.1243±0.035	49.8553±0.031	0.7415±0.017	0.2336±0.026	0.1222±0.008	1.5765±0.01	**1.0**±0.0	1.6085±0.0
MORFITT 🇫🇷	$-\mathcal{D}$	DNN	49.7506±0.009	59.038±0.012	0.6499±0.022	0.2323±0.021	0.0398±0.005	2.0748±0.015	0.796±0.045	2.4454±0.003
	$-\mathcal{D}$	DC	55.4551±0.01	62.5306±0.008	0.6134±0.003	0.2243±0.003	0.0425±0.001	2.0332±0.006	0.8775±0.014	2.4411±0.001
	$-\mathcal{D}$	MCD	48.3269±0.008	57.3529±0.008	0.6309±0.021	0.1519±0.05	0.0464±0.007	2.2692±0.03	**0.9856**±0.006	2.4767±0.003
	$-\mathcal{D}$	VI	53.0834±0.014	61.6728±0.01	0.6408±0.042	0.2571±0.039	0.0477±0.006	**2.0245**±0.007	0.7724±0.047	**2.4369**±0.004
	$+\mathcal{D}$	DNN	53.4963±0.019	61.8015±0.014	0.6081±0.017	0.2098±0.014	0.0363±0.002	2.0538±0.015	0.8334±0.01	2.4453±0.002
	$+\mathcal{D}$	DC	**56.4418**±0.018	**62.9596**±0.02	0.6148±0.027	0.2325±0.018	0.0433±0.003	2.0251±0.015	0.8667±0.03	2.4394±0.001
	$+\mathcal{D}$	MCD	51.8519±0.015	60.5392±0.006	0.5718±0.003	0.0687±0.022	0.0298±0.0	2.1426±0.01	0.9651±0.005	2.4629±0.002
	$+\mathcal{D}$	VI	54.2993±0.011	62.7145±0.01	**0.5346**±0.008	**0.0488**±0.018	**0.0279**±0.002	2.1064±0.014	0.9752±0.007	2.4602±0.002

Table 5: Comparison for text classification performance and uncertainty-awareness. We report the mean of 10 seed runs for all the metrics. We denote best score with **bold** and second best with underline. We denote both English and French domain-specific PLMs with $+\mathcal{D}$. The models DC, MCD, VI are from the $+\mathcal{U}$ set.

Can Rule-Based Insights Enhance LLMs for Radiology Report Classification? Introducing the RadPrompt Methodology.

Panagiotis Fytas[1] **Anna Breger**[*,2,3] **Ian Selby**[*,4,5]
Simon Baker[1] **Shahab Shahipasand**[5] **Anna Korhonen**[1]

[1]Language Technology Lab, University of Cambridge
[2]Department of Applied Mathematics and Theoretical Physics, University of Cambridge
[3]Center of Medical Physics and Biomedical Engineering, Medical University of Vienna
[4]Department of Radiology, University of Cambridge
[5]Cambridge University Hospitals, NHS Foundation Trust
pf376@cam.ac.uk

Abstract

Developing imaging models capable of detecting pathologies from chest X-rays can be cost and time-prohibitive for large datasets as it requires supervision to attain state-of-the-art performance. Instead, labels extracted from radiology reports may serve as distant supervision since these are routinely generated as part of clinical practice. Despite their widespread use, current rule-based methods for label extraction rely on extensive rule sets that are limited in their robustness to syntactic variability. To alleviate these limitations, we introduce RadPert, a rule-based system that integrates an uncertainty-aware information schema with a streamlined set of rules, enhancing performance. Additionally, we have developed RadPrompt, a multi-turn prompting strategy that leverages RadPert to bolster the zero-shot predictive capabilities of large language models, achieving a statistically significant improvement in weighted average F1 score over GPT-4 Turbo. Most notably, RadPrompt surpasses both its underlying models, showcasing the synergistic potential of LLMs with rule-based models. We have evaluated our methods on two English Corpora: the MIMIC-CXR gold-standard test set and a gold-standard dataset collected from the Cambridge University Hospitals.

1 Introduction

Supervised deep learning for medical imaging classification has accomplished significant milestones. In the chest X-ray (CXR) domain, such models have exhibited predictive capabilities on par with expert physicians (Rajpurkar et al., 2018; Tang et al., 2020) and are being utilized in collaborative settings to increase clinician accuracy (Rajpurkar et al., 2020).

Annotating medical images, however, is expensive and arduous: it requires a committee of expert radiologists to resolve the inherently high degree of annotator variance and subjectivity (Razzak et al., 2018). This issue is particularly problematic considering the global shortage of radiologists (Jeganathan, 2023; Kalidindi and Gandhi, 2023; Konstantinidis, 2023). Instead, we often have access to a form of distant supervision: the radiology report. Radiology reports are semi-structured free-text interpretations of an X-ray image and are generated as a routine part of clinical practice to communicate findings.

In the past, rule-based models (Irvin et al., 2019; Peng et al., 2017) have been used to extract structured labels from radiology reports in various imaging datasets, including ChestX-ray14 (Wang et al., 2017), CheXpert (Irvin et al., 2019), MIMIC-CXR (Johnson et al., 2019) and BRAX (Reis et al., 2022). However, those rule-based methods are often based on elementary techniques and, thus, exhibit limited robustness to syntactic variation. Naturally, supervised deep learning models offer superior performance through their robustness to syntactic variability (Smit et al., 2020; Jain et al., 2021b). In contrast, Large Language Models (LLMs) represent a significant improvement over rule-based models in an unsupervised setting and have achieved impressive performance in the field of radiology (Infante et al., 2024; Adams et al., 2023; Liu et al., 2023).

In this paper, we present RadPert, a rule-based model built on the RadGraph knowledge graph (Jain et al., 2021a). RadPert leverages entity-level uncertainty labels from RadGraph, reducing the

[*]Equal contribution.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 212–235
August 16, 2024. ©2024 Association for Computational Linguistics

need for a comprehensive rule set and enhancing its resilience to syntactic variations. We have evaluated RadPert internally on MIMIC-CXR and externally on a dataset collected from the Cambridge University Hospitals (CUH). RadPert surpasses CheXpert, the former rule-based state-of-the-art (SOTA), by achieving statistically significant improvement in weighted average F1 score.

Furthermore, we explore the collaborative potential of LLMs with rule-based models through RadPrompt. RadPrompt is a multi-turn prompting strategy that employs RadPert as an implicit means of encoding medical knowledge (Figure 1). In fact, RadPrompt, based on GPT-4 Turbo, manages to outperform both its underlying models in a zero-shot setting.

2 Related Work

Numerous natural language processing methods have been developed to derive structured predictions from radiology reports (Peng et al., 2017; Hassanpour et al., 2017; Pons et al., 2016; Bozkurt et al., 2019; Wang et al., 2018). Many of those approaches are designed for the multitask classification of radiology reports, written in English, into labels representing prevalent pathologies from CXRs. Each such label can exhibit one of four output classes: *Null, Positive, Negative* and *Uncertain*. CheXpert (Irvin et al., 2019), the rule-based SOTA, follows an approach based on regular expression matching and the Universal Dependency Graph (UDG) of a radiology report. Due to the rudimentary regular expression matching, however, CheXpert is sensitive to syntactic variation. Thus, multiple over-generalized rules are used in an attempt to alleviate these shortcomings. Furthermore, the UDG is a type of information extraction that does not explicitly identify negation and uncertainty. Therefore, its ability to detect uncertainty in complex phrases is hampered despite the extensive rule set. Extensions of CheXpert have been developed for Brazilian Portuguese (Reis et al., 2022) and German (Wollek et al., 2024). CheXbert (Smit et al., 2020) is a semi-supervised model pretrained on automatically extracted labels from the CheXpert model, fine-tuned on manually annotated reports, and evaluated on 687 MIMIC-CXR gold-standard test set reports. However, the published model weights[1] of CheXbert differ from the original model. This discrepancy complicates comparisons on the MIMIC-CXR dataset as the published model is fine-tuned on unspecified MIMIC-CXR manually annotated reports, which can potentially overlap with the MIMIC-CXR gold-standard test set.

Recent work has also explored the adoption of LLMs for radiology report classification. Specifically, Dorfner et al. (2024) examine the zero and few-shot capabilities of LLMs. However, they mainly treat the task as a binary classification for each pathology. Namely, for multitask classification, they only report the few-shot results on an unpublished institutional dataset. CheX-GPT (Gu et al., 2024) utilizes zero-shot GPT-4 labels as a distant supervision to fine-tune a BERT-based model. Nonetheless, they also simplify the task into binary classification.

Alternative approaches to the classification of chest X-rays (CXRs) explore moving away from the distantly supervised paradigm of training unimodal vision models on classifying structured labels extracted from radiology reports. In lieu of structured prediction, Vision-Language (VL) models are trained to align the embedding representations of CXRs with the representations of the corresponding radiology reports via self-supervised contrastive learning objectives (Huang et al., 2021; Boecking et al., 2022; Tiu et al., 2022; Wang et al., 2022; Bannur et al., 2023). This alignment task is transformed into CXR classification through the cosine similarity of CXR embeddings to the embeddings of textual prompts representing the existence or absence of pathologies. However, vision models trained with the structured prediction paradigm outperform VL models such as CheXzero (Tiu et al., 2022), even when the latter utilizes an expert-annotated validation set for selecting optimal classification thresholds.

In this paper, we will focus on improving the unsupervised SOTA for the multitask classification of radiology reports.

3 Methods

3.1 Task

Similar to CheXpert and CheXbert, we will focus on the multitask classification of CXR radiology reports. Specifically, our models classify thirteen labels that correspond to pathologies (Atelectasis, Edema, Cardiomegaly, Consolidation, Enlarged Cardiomediastinum, Fracture, Lung Lesion, Lung Opacity, Pleural Effusion, Pleural Other, Pneumoth-

[1] https://github.com/stanfordmlgroup/CheXbert

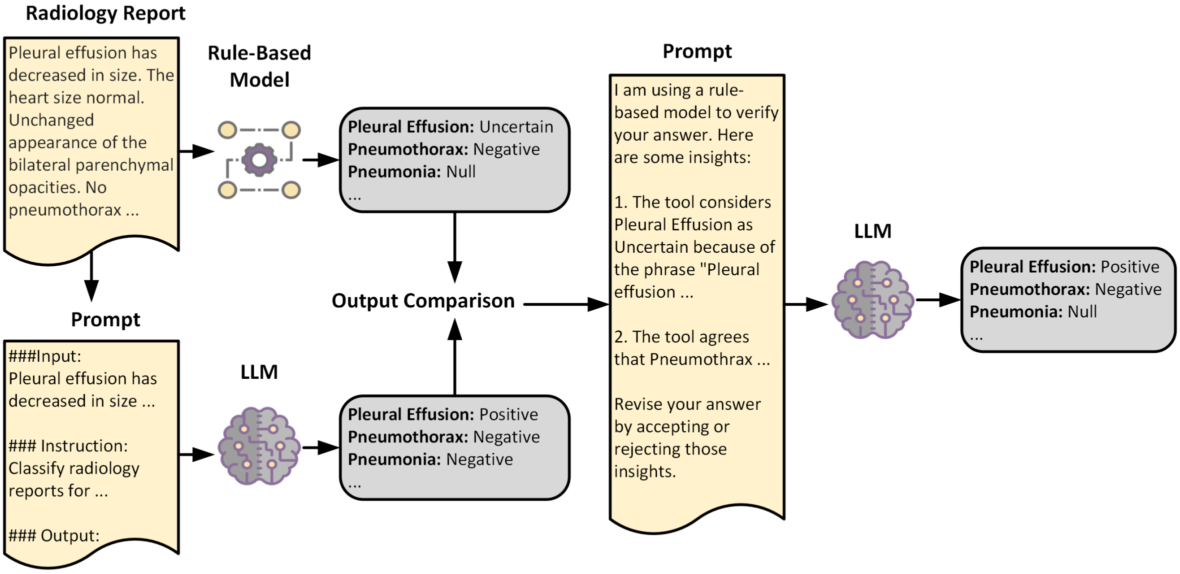

Radiology Report
Pleural effusion has decreased in size. The heart size normal. Unchanged appearance of the bilateral parenchymal opacities. No pneumothorax ...
Rule-Based Model
Pleural Effusion: Uncertain
Pneumothorax: Negative
Pneumonia: Null
...
Output Comparison
Prompt
###Input:
Pleural effusion has decreased in size ...
### Instruction:
Classify radiology reports for ...
### Output:
LLM
Pleural Effusion: Positive
Pneumothorax: Negative
Pneumonia: Negative
...
Prompt
I am using a rule-based model to verify your answer. Here are some insights:
1. The tool considers Pleural Effusion as Uncertain because of the phrase "Pleural effusion ...
2. The tool agrees that Pneumothrax ...
Revise your answer by accepting or rejecting those insights.
LLM
Pleural Effusion: Positive
Pneumothorax: Negative
Pneumonia: Null
...

Figure 1: Overview of the RadPrompt methodology. RadPrompt utilizes the rule-based RadPert model to detect potential errors in the original (first-turn) LLM classification decision. A second-turn prompt is then constructed, offering evidence that may cause the LLM to revise its original classification outcome.

orax, Support Devices and Pneumonia), with each label having four possible output classes: *Null, Positive, Negative* and Uncertain. A pathology is classified as *Null* if there are no references to it in the radiology report. It is considered *Negative* when its absence is explicitly mentioned. *Positive* classes entail that the existence of the corresponding pathology is specified in the report. Finally, *Uncertain* classes imply that while the pathology is discussed in the report, its existence cannot be determined.

3.2 RadPert

In order to overcome the limitations of existing tools, we have designed RadPert. RadPert incorporates hand-crafted rules with the RadGraph (Jain et al., 2021a) knowledge graph.

3.2.1 RadGraph Information Schema

RadGraph (Jain et al., 2021a) defines an information schema specifically designed for radiology reports. It contains two top-level entity types: *Anatomy (ANAT)* and *Observation (OBS)*. *Anatomy* entities describe bodily anatomical structures (e.g. “lobe”) and their spatial characteristics (e.g. “left”). *Observation* entities include pathological abnormalities (e.g. “opacities”), diagnosed diseases (e.g. “pneumonia”) and various other characteristics (e.g. “acute”). It is important to note that *Observation* entities are further categorized into three second-level attributes: *Definitely Present (DP)*, *Definitely Absent (DA)* and *Uncertain (U)*.

Additionally, RadGraph defines three types of directed relations between entities. Firstly, the *suggestive of* relation indicates that some *Observation* implies the existence of another *Observation*. Secondly, *located at* relations account for *Observations* relating to specific *Anatomies*. Finally, *modify* relations can exist only between the same type of entity and describe the characteristics relating to a specific entity (e.g., *modify*(“left”, “lung”)).

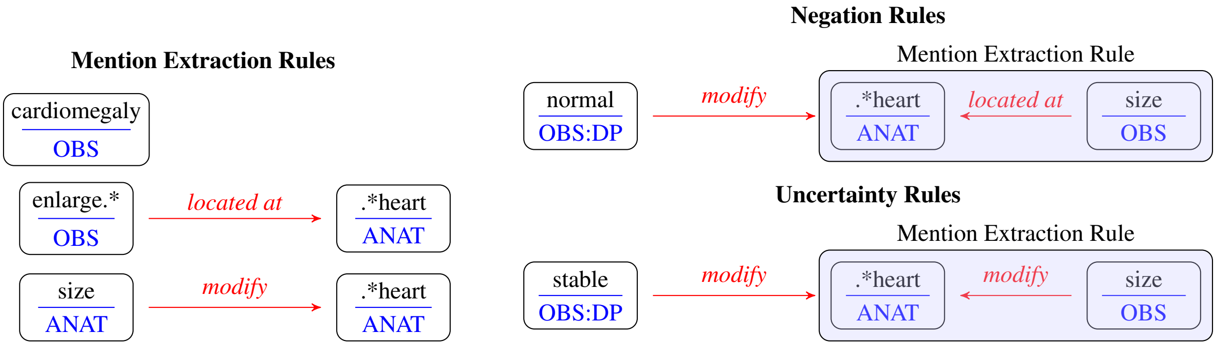
The RadGraph model is based on the DyGIE++ (Wadden et al., 2019) framework initialized with PubMedBERT weights (Gu et al., 2021). The model is fine-tuned on 500 expert-annotated MIMIC-CXR reports based on the RadGraph information schema.

3.2.2 RadPert Pipeline

RadPert employs the following four-stage pipeline:

Knowledge graph extraction. We first extract the RadGraph entities and relations from radiology reports. Utilizing RadGraph instead of the UDG allows uncertainty and negation classes to be extracted at an entity level. Thus, the negation and the uncertainty of various complex phrases can be determined based on those classes, reducing the need for complex negation and uncertainty rules.

Mention extraction. In this stage, for each pathology label, we have adapted and simplified the CheXpert rules (Irvin et al., 2019) so they can be applied to RadGraph entities and relations. Essentially, those rules can be represented as graphs



(a) *Mention extraction* rules.

(b) *Negation/uncertainty detection* rules.

Figure 2: Examples of RadPert rules for Cardiomegaly. The rules take the form of graphs that follow the RadGraph (Jain et al., 2021a) information schema. The ".*" symbolizes allowing the matching of different prefixes and suffixes within the entity span.

based on the RadGraph information schema. Figure 2a includes examples of mention extraction rules in the form of graphs. Checking whether a pathology is mentioned in a radiology report amounts to determining whether any rule-graphs for the specific pathology are subgraphs of the radiology report knowledge graph[2]. If none of the pathology rules match a given radiology report, then the class for that pathology is *Null*.

Negation/uncertainty detection. We next aim to determine whether an extracted mention is *Positive*, *Negative*, or *Uncertain*. For mentions that contain *Observation* entities in their subgraph, the uncertainty quantifier of the *Observation* determines the initial class of that mention. For instance, if a "heart" *Anatomy* is connected with an "enlarged" *Observation*, which is characterized as *Definitely Absent*, then that mention will be labeled as *Negative*. If a mention only possesses *Anatomy* entities, then we consider by default that mention to be *Positive*. However, certain phrases contain implicit negations/uncertainties. In cases such as "normal heart size", the entity "normal" would be considered under RadGraph a *Definitely Present Observation* attached to an *Anatomy*. Thus, in order to detect such implicit negations/uncertainties and determine the final uncertainty class for each pathology, we have developed a negation and an uncertainty rule set. Both rule sets are constructed from hand-crafted rules in the form of graphs. Examples of Cardiomegaly negation/uncertainty rules can be observed in Figure 2b. When a negation rule is activated, the initial class of the mention will be negated (i.e., *Positive* becomes *Negative* and *Negative* becomes *Positive*). However, when an uncertainty rule is matched, RadPert considers the class of the mention to be *Uncertain*.

Mention aggregation. After extracting and classifying all mentions in a radiology report for a specific label, RadPert aggregates them into the final uncertainty class for that label. Similarly to CheXpert (Irvin et al., 2019), we prioritize positive mentions, followed by uncertain ones, while negative mentions have the lowest priority.

3.3 RadPrompt

RadPert, through its rules, implicitly encodes expert knowledge vital to classifying radiology reports. However, as a rule-based system, it is still sensitive to syntactic and lexical variability. To alleviate this limitation, we propose RadPrompt, a zero-shot prompting technique that injects prompts with insights derived from the application of RadPert. RadPrompt, as seen in Figure 1, employs a two-turn prompting strategy.

In the first turn, the zero-shot prompt contains instructions, which define the task, and the radiology report that needs to be classified. After a response is received from the LLM, the first-turn classification outcome is compared with the output of RadPert.

In the second turn, a prompt is constructed by specifying that a rule-based model is used to verify the validity of the LLM's answer. Hints are then added by specifying for each pathology either RadPert's agreement with the LLM or the radiology report sentence that leads RadPert to a disagreement. This is possible since RadPert, as a rule-based sys-

[2] This problem corresponds to the edge-colored and node-colored variant of Induced Subgraph Isomorphism. Exhaustive search with subgraphs of fixed-length has polynomial complexity (Floderus et al., 2015).

tem, allows the detection of the specific mention that leads to the classification decision. Finally, the prompt instructs the LLM to adjust its answer by accepting or rejecting RadPert's hints. In Table 14 of the Appendix, we present the format of our first and second-turn prompts.

3.3.1 Base Model

As a base model for the RadPrompt strategy, we explore various LLMs, including API-based models such as Gemini-1.5 Pro (Reid et al., 2024), Claude-3 Sonnet, GPT-4 Turbo (OpenAI, 2023), and Llama-2 (Touvron et al., 2023). In the case of Llama-2, we are using the 70 billion parameter chat variant, quantized with the Int 4 AWQ method (Lin et al., 2024), which we run locally with a single NVIDIA RTX 6000 Ada GPU.

4 Results and Discussion

4.1 Evaluation

To allow comparison with previous work (Irvin et al., 2019; Smit et al., 2020), for each pathology, we evaluate our methodology based on the weighted average F1 score across three aspects of the task: negation detection, positive mention detection, and uncertainty detection. We report the F1 scores of the sub-tasks in the Appendix. Each of those sub-tasks amounts to binary classification. For instance, *Negative* classes are transformed into positive in negation detection, while the other classes are transformed into negative. Positive mention detection and uncertainty detection are constructed with an analogous logic. The reported scores correspond to the averages across 1000 bootstrap replicates (Efron and Tibshirani, 1986), reported along the 95% Confidence Intervals (CI).

4.2 Data

For internal evaluation, we are evaluating the models on the gold-standard test set of annotated radiology reports used in the MIMIC-CXR paper (Johnson et al., 2019). MIMIC-CXR is considered an internal dataset for methods based on RadPert since RadGraph is trained on MIMIC-CXR radiology reports. The MIMIC-CXR gold-standard test set contains 687 radiology reports that do not overlap with the training and validation set of RadGraph.

For external evaluation, we have collected a private dataset from the Cambridge University Hospitals in Cambridge, UK. The CUH dataset consists of 650 radiology reports annotated by a single consultant radiologist with six years of experience, using the same annotation guidelines as MIMIC-CXR[3]. Details regarding the label distribution of both datasets are attached in Table 15 of the Appendix.

4.3 RadPert Evaluation

In Table 1, we report the weighted average F1 scores across the sub-tasks of positive mention detection, negation detection, and uncertainty Detection for the MIMIC-CXR and CUH datasets. We are also reporting the improvements over the CheXpert labeler alongside their confidence intervals. Radpert achieves a statistically significant improvement both on average and on the majority of the pathologies. Namely, for MIMIC-CXR, RadPert is 8.0% (95% CI: 5.5%, 10.8%) better than CheXpert, yielding an average F1 score of 0.757 (95% CI: 0.779, 0.800).

In Table 6 of the Appendix, we also report fine-grained results in the distinct sub-tasks. In addition to the sub-tasks of negation, positive mention, and uncertainty detection, we also report the performance improvement in mention detection. Mention detection treats *Null* as the positive class, and *Negative*, *Uncertain*, and *Positive* as the negative class.

4.3.1 Discussion on RadPert's Performance

We observe performance improvement in all subtasks. The strongest improvement is achieved in the uncertainty detection task, showcasing the effectiveness of utilizing the uncertainty labels of RadGraph. However, the improvement in mention detection is marginal. A primary cause of mention detection failure is the reliance on the RadGraph model, which occasionally fails to recall all entities and relations within a radiology report.

Focusing on specific pathologies, RadPert fails to consistently outperform CheXpert for Atelectasis, Edema, and Pleural Effusion. In the case of Atelectasis and Edema, the rule sets are straightforward, and their mentions often lack syntactic variability in practice, offering limited benefit from the uncertainty-aware entity representations of RadGraph. Regarding Pleural Effusion, RadPert is hindered by the divergence between RadGraph annota-

[3] MIMIC-CXR annotation guidelines were provided upon request by the authors of Johnson et al. (2019).

Pathologies	MIMIC-CXR Gold Standard Test Set				CUH			
	Weighted F1 RadPert		Improvement over CheXpert (%)		Weighted F1 RadPert		Improvement over CheXpert (%)	
Atelectasis	0.782	(0.740, 0.825)	-5.2	(-10.2, 0.2)	0.893	(0.836, 0.941)	-0.8	(-6.3, 4.4)
Cardiomegaly	0.801	(0.749, 0.846)	8.1	(4.2, 12.6)	0.910	(0.872, 0.945)	27.3	(16.4, 41.1)
Consolidation	0.806	(0.731, 0.872)	15.5	(1.9, 33.4)	0.951	(0.928, 0.971)	3.0	(0.4, 5.8)
Edema	0.801	(0.758, 0.843)	0.1	(-5.6, 3.9)	0.625	(0.466, 0.754)	-5.5	(-28.1, 19.1)
Enlarged Card.	0.628	(0.548, 0.702)	23.8	(5.6, 4.7)	0.908	(0.860, 0.950)	0.7	(-2.1, 3.5)
Fracture	0.866	(0.765, 0.946)	30.8	(9.7, 60.9)	0.764	(0.593, 0.898)	12.7	(-8.1, 47.5)
Lung Lesion	0.696	(0.583, 0.797)	4.0	(-5.4, 14.8)	0.816	(0.706, 0.911)	660.4	(210.8, 2700.3)
Lung Opacity	0.783	(0.741, 0.827)	3.2	(-1.3, 8.7)	0.712	(0.652, 0.766)	0.8	(-1.6, 3.5)
Pleur. Effusion	0.873	(0.843, 0.901)	-3.3	(-6.4, -0.2)	0.641	(0.587, 0.689)	0.1	(-2.5, 2.8)
Pleur. Other	0.547	(0.390, 0.692)	16.7	(1.6, 44.0)	0.082	(0.043, 0.127)	189.8	(45.9, 713.3)
Pneumonia	0.757	(0.704, 0.806)	28.1	(15.8, 42.5)	0.656	(0.520, 0.773)	54.5	(9.4, 130.9)
Pneumothorax	0.898	(0.856, 0.934)	5.1	(-0.4, 10.9)	0.626	(0.568, 0.682)	2.1	(-0.9, 5.7)
Sup. Devices	0.886	(0.854, 0.915)	2.1	(-0.4, 5.1)	0.858	(0.825, 0.890)	-2.6	(-4.8, -0.6)
Macro Avg.	0.757	(0.779, 0.800)	8.0	(5.5, 10.8)	0.726	(0.699, 0.752)	14.6	(10.4, 19.1)
Weighted Avg.	0.816	(0.802, 0.830)	3.4	(1.5, 5.3)	0.787	(0.765, 0.808)	5.0	(2.6, 7.3)

Table 1: Weighted average F1 scores for RadPert alongside improvements over the CheXpert model on the MIMIC-CXR gold-standard and CUH test sets. The F1 scores are averaged across the sub-tasks of positive mention detection, negation detection, and uncertainty detection weighted by the support sets. The scores correspond to the averages across 1000 bootstrap replicates and are reported alongside their confidence intervals.

tion guidelines[4] and those of the MIMIC-CXR and CUH datasets concerning uncertainty. Specifically, RadGraph suggests annotating any degree of uncertainty as *OBS:Uncertain* (Jain et al., 2021a) while the MIMIC-CXR guidelines, also used by CUH, permit some degree of uncertainty within *Positive* and *Negative* labels. For instance, "likely representing pneumonia" should be labeled as positive according to MIMIC-CXR guidelines. For Pleural Effusion, uncertain mentions such as "minimal if any pleural effusion" are commonplace and labeled inconsistently by the annotators in MIMIC-CXR. However, due to RadGraph's annotation guidelines, RadPert primarily labels such mentions as *Uncertain*, resulting in low precision in the uncertainty detection task for Pleural Effusion. This behavior can be observed in the Pleural Effusion confusion matrices (Appendix, Figure 3).

Notably, RadPert's performance for Lung Lesion showed a substantial improvement over CheXpert's performance on the CUH dataset compared to MIMIC-CXR. This discrepancy arises because "lung lesion" is a specific term frequently used in the CUH reports, while it rarely appears in MIMIC-CXR reports. The CheXpert labeler treats Lung Lesion as an umbrella term encompassing "masses", "nodular opacities", and "carcinomata", lacking specific rules for "lung lesions" and only identifying the less general terms, leading to inconsistent performance in CUH. Additionally, variations such as "edema" in the US and "oedema" in the UK also illustrate the divergent terminology and spelling conventions between the two corpora, although these spelling differences do not affect the ability of CheXpert to detect Edema mentions.

Finally, in Table 5 of the Appendix, we provide carbon estimates for both CheXpert and RadPert. RadPert not only improves upon CheXpert in performance but also demonstrates greater energy efficiency.

4.4 RadPrompt Evaluation

In Table 2, we present the improvement in the weighted average F1 score of RadPrompt for various base LLMs on the MIMIC-CXR gold-standard test set. Specifically, we compare the revised classification outcome of the second-turn prompt, which is infused with RadPert hints, to the first-turn classification outcome. For all tested LLMs, we observe that the RadPrompt strategy leads, on average (across pathologies), to a statistically significant improvement over the baseline zero-shot prompting. For clarity, in Tables 7, 8, 9, 10 and 11 of the Appendix, we also report the task-specific F1 scores of the first and second turns of RadPrompt.

Furthermore, we compare RadPrompt's second-

[4] Available on OpenReview.

	RadPrompt Improvement of Weighted Average F1 Over 1st Turn (%)							
Pathologies	**Gemini-1.5 Pro**		**Llama-2 70B**		**Claude-3 Sonnet**		**GPT-4 Turbo**	
Atelectasis	-0.9	(-4.4, 3.0)	-7.0	(-12.6, -0.2)	-1.4	(-7.1, 5.3)	-3.9	(-7.2, -0.4)
Cardiomegaly	-2.3	(-6.6, 1.9)	14.3	(9.2, 20.2)	2.7	(-2.4, 7.6)	-1.9	(-5.5, 1.5)
Consolidation	26.6	(13.9, 40.5)	70.7	(43.8, 102.6)	31.9	(15.7, 49.7)	2.6	(-3.6, 9.4)
Edema	7.7	(3.2, 12.5)	10.3	(4.5, 16.6)	7.4	(1.9, 13.1)	-3.1	(-5.9, -0.4)
Enlarged Card.	49.7	(22.1, 89.6)	160.2	(75.1, 309.4)	103.0	(55.7, 167.3)	3.9	(-8.6, 17.3)
Fracture	10.7	(1.4, 23.6)	20.1	(4.6, 42.0)	14.8	(0.8, 31.2)	5.2	(0.9, 9.9)
Lung Lesion	65.5	(37.3, 100.6)	24.0	(3.7, 48.0)	3.2	(-11.5, 18.5)	6.5	(-7.0, 20.4)
Lung Opacity	26.9	(18.8, 36.2)	23.5	(15.9, 32.3)	23.6	(14.1, 34.0)	8.1	(2.2, 14.4)
Pleural Effusion	4.1	(1.5, 6.5)	4.9	(1.2, 9.0)	8.3	(5.2, 11.4)	0.3	(-1.8, 2.4)
Pleural Other	21.0	(1.8, 44.6)	158.3	(-0.1, 291.8)	36.8	(8.2, 72.8)	10.8	(-6.9, 29.4)
Pneumonia	15.6	(10.3, 21.4)	-5.3	(-14.1, 4.0)	22.0	(14.2, 30.5)	4.5	(1.2, 8.3)
Pneumothorax	20.5	(14.9, 26.3)	19.3	(12.7, 26.8)	34.9	(28.2, 42.5)	1.0	(-1.3, 3.3)
Support Devices	4.1	(1.8, 6.7)	23.1	(15.7, 31.7)	1.1	(-0.8, 3.3)	0.5	(-0.5, 1.6)
Macro Average	14.8	(12.2, 17.3)	20.8	(16.2, 25.8)	16.2	(13.1, 19.4)	2.1	(0.3, 4.1)
Weighted Average	10.2	(8.4, 12.0)	12.5	(9.7, 15.4)	12.7	(10.7, 15.0)	0.9	(-0.2, 2.1)

Table 2: Improvement of weighted average F1 scores for RadPrompt over the base LLM on MIMIC-CXR gold-standard test set, alongside confidence intervals. Improvement is measured in a multi-turn chat setting by comparing the initial classification decision of the LLM to the revised classification decision after introducing RadPert hints.

	RadPrompt Improvement of Weighted Average F1 Over RadPert (%)							
Pathologies	**Gemini-1.5 Pro**		**Llama-2 70B**		**Claude-3 Sonnet**		**GPT-4 Turbo**	
Atelectasis	6.2	(0.8, 11.7)	-0.0	(-1.6, 1.7)	3.8	(0.6, 7.5)	6.2	(0.7, 11.7)
Cardiomegaly	-1.4	(-4.0, 1.2)	0.7	(-0.9, 2.4)	-0.2	(-1.5, 1.0)	0.7	(-2.8, 4.5)
Consolidation	-7.7	(-16.0, 0.1)	-22.4	(-29.6, -16.2)	-0.6	(-4.2, 3.2)	2.4	(-3.8, 9.1)
Edema	-0.9	(-3.9, 2.3)	0.5	(-0.8, 1.9)	0.1	(-0.8, 1.2)	1.3	(-1.7, 4.7)
Enlarged Card.	-11.6	(-19.1, -5.1)	-1.5	(-4.0, 0.7)	-8.0	(-14.1, -2.4)	-6.5	(-12.8, -0.8)
Fracture	-8.5	(-15.7, -1.2)	-1.2	(-4.0, 1.2)	-2.0	(-5.2, 0.0)	-4.5	(-11.7, 3.3)
Lung Lesion	-2.4	(-9.2, 4.9)	-28.4	(-37.9, -19.4)	2.1	(-5.3, 11.1)	-2.9	(-14.0, 9.2)
Lung Opacity	-5.0	(-8.0, -2.1)	-0.4	(-1.9, 1.1)	-0.4	(-2.7, 1.8)	-0.2	(-3.1, 2.8)
Pleural Effusion	2.0	(0.0, 4.1)	-0.7	(-2.1, 0.9)	3.2	(1.6, 5.0)	2.8	(0.4, 5.4)
Pleural Other	-10.0	(-20.3, 1.8)	-4.0	(-12.5, 0.0)	0.0	(0.0, 0.0)	13.5	(-3.9, 39.7)
Pneumonia	4.2	(-0.1, 9.4)	-14.8	(-19.8, -9.7)	3.0	(0.5, 6.4)	4.4	(-0.4, 9.5)
Pneumothorax	-0.6	(-3.1, 2.1)	-3.0	(-5.0, -1.3)	2.7	(0.3, 5.6)	3.5	(0.8, 7.1)
Support Devices	2.2	(0.5, 4.0)	-0.2	(-1.2, 0.5)	1.2	(-0.0, 2.5)	0.2	(-2.4, 2.8)
Macro Average	-2.2	(-3.8, -0.6)	-5.5	(-6.9, -4.3)	0.5	(-0.4, 1.4)	1.4	(-0.5, 3.2)
Weighted Average	-0.2	(-1.5, 1.2)	-3.5	(-4.4, -2.7)	1.4	(0.7, 2.1)	1.9	(0.7, 3.2)

Table 3: Improvement of weighted average F1 scores for RadPrompt over the rule-based RadPert on the MIMIC-CXR gold-standard test set, alongside confidence intervals.

turn results with RadPert in Table 3 for the MIMIC-CXR gold-standard test set. On average, RadPrompt with Gemini-1.5 Pro and Llama-2 70 B fail to outperform RadPert. However, Claude-3 Sonnet and GPT-4 Turbo-based RadPrompt surpass RadPert.

Regarding the external evaluation of RadPrompt, the current ethical agreement with the Cambridge University Hospitals limits the use of third-party APIs. Thus, we are only able to evaluate RadPrompt with a Llama-2 base. We present the weighted average and the sub-task-specific results in Tables 12 and 13. Similarly to the MIMIC-CXR gold-standard test set, we observe that Llama-2-based RadPrompt enhances the performance of Llama-2 but fails to improve upon RadPert.

4.4.1 Discussion on RadPrompt's Performance

We can observe from Tables 2 and 3 that RadPrompt on Claude-3 Sonnet and on GPT-4 Turbo exceeds, on average, both RadPert and the initial LLM predictions. Namely, RadPrompt with GPT-4 Turbo is 2.1% (CI 0.3%, 4.1%) better than baseline GPT-4 Turbo and 1.4% (CI -0.5%, 3.2%) better than RadPert.

Focusing on individual pathologies, we notice that RadPrompt with a Gemini-1.5 Pro base manages to outperform both of its underlying models for Pleural Effusion, Pneumonia, and Support Devices. Additionally, RadPrompt with Claude-3 Sonnet surpasses its underlying models in the case of Lung Lesion, Pleural Effusion, Pneumonia, Pneumothorax, and Support Devices. For a GPT-4 Turbo base, the same behavior is observed for Consolidation, Pleural Effusion, Pleural Other, Pneumonia, and Pneumothorax. The ability of RadPrompt to boost the performance of both its underlying models demonstrates the potential of combining the language reasoning capabilities of LLMs with the insights encoded in rule-based models.

In Table 4, we present a fine-grained comparison between the first and second turns of RadPrompt. We observe that all models, with the exception of GPT-4 Turbo, initially struggled to understand that we intended to classify only those pathologies explicitly mentioned in the report. This effect disproportionately affects the *Negative* class since *Null* is often conflated with *Negative*. The distinction, however, between those two labels is non-negligible. Inconsistencies often exist between the gold-standard labels extracted directly from chest X-ray Images and the gold-standard labels of their corresponding radiology reports, and thus, pathologies visible within a chest X-ray may be excluded from the radiology report (Jain et al., 2021b). Such observations are also noted in other clinical domains, such as Magnetic Resonance Imaging (MRI), where the clinical context and the referrer physician may bias the observations mentioned within a radiology report (Wood et al., 2020).

5 Limitations

While this study demonstrates promising improvements in radiology report classification using the RadPrompt methodology, several limitations must be considered.

RadPert and RadPrompt are exclusively developed and tested for the English language. The study also centers around a list of pathologies typical of chest X-rays. As such, the extension of our methodologies to other languages, types of medical imaging, and additional pathologies was not verified.

Furthermore, previous studies have highlighted discrepancies between labels from radiology report annotations and those from the corresponding imaging study annotations (Jain et al., 2021b; Wood et al., 2020). The source of such inconsistencies includes incomplete radiology report impressions, hierarchical relationships within labels, and the undeniable uncertainty of the task. In future work, we aim to study this effect within the CUH test set.

Due to ethical considerations, we are currently unable to perform inference for the CUH test set through third-party APIs. Thus, we have not evaluated RadPrompt externally for SOTA LLMs. We expect to overcome this limitation after the planned release of the CUH dataset.

Additionally, we cannot estimate the computational cost and carbon footprint for GPT-4-based RadPrompt due to a lack of specific metrics. In the Appendix, we provide carbon footprint estimates for the Llama-2-based RadPrompt, which is significantly higher than RadPert and CheXpert. Nonetheless, RadPert delivers performance comparable to GPT-4 while operating on a commercial CPU with minimal carbon emissions, underscoring its benefits in resource-limited environments.

Finally, there is an inherent degree of ambiguity in classifying radiology reports, especially as it pertains to the *Uncertainty* labels. We aim to extend current datasets with labels from multiple annota-

	RadPrompt Improvement of Weighted Average F1 Over 1st Turn (%)			
Sub-task	**Gemini-1.5 Pro**	**Llama-2 70B**	**Claude-3 Sonnet**	**GPT-4 Turbo**
Mention Detection	17.8 (15.6, 20.0)	26.7 (23.8, 29.8)	24.6 (21.7, 27.8)	1.9 (0.9, 3.0)
Negation Detection	31.9 (26.4, 37.6)	54.8 (45.8, 64.2)	62.3 (52.1, 73.1)	4.9 (2.3, 8.1)
Pos. Mention Detection	3.8 (2.4, 5.2)	1.7 (-0.5, 4.1)	0.7 (-0.9, 2.4)	-0.4 (-1.6, 0.7)
Uncertainty Detection	2.9 (-5.9, 13.0)	-6.4 (-20.7, 9.8)	-0.5 (-13.0, 14.0)	-2.6 (-10.3, 5.9)
Weighted Average	10.2 (8.4, 12.0)	12.5 (9.7, 15.4)	12.7 (10.7, 15.0)	0.9 (-0.2, 2.1)

Table 4: Improvement of RadPrompt over the base LLM for the different sub-tasks on MIMIC-CXR gold-standard test set. For each sub-task. we report the improvement of the weighted average F1 score across all pathologies, along with confidence intervals. The weighted average refers to averaging over sub-tasks, excluding the mention detection task.

tors in order to measure annotator agreement.

6 Conclusions

This paper introduced RadPert, a rule-based system enhanced by the RadGraph information schema, demonstrating significant improvements in the classification of radiology reports. By leveraging entity-level uncertainty labels, RadPert reduces reliance on comprehensive rule sets. Our evaluations show that RadPert surpasses CheXpert, the previous rule-based SOTA, by achieving an 8.0% (95% CI: 5.5%, 10.8%) increase in F1 score, with confidence intervals strongly supporting this improvement.

Further extending the application of RadPert, we developed RadPrompt, a multi-turn prompting strategy that utilizes insights from RadPert to enhance the zero-shot prediction capabilities of large language models. RadPrompt demonstrated a 2.1% (95% CI: 0.3%, 4.1%) improvement in F1 score over GPT-4 Turbo, indicating its potential to refine predictions in clinical settings. These results highlight the growing synergy between structured rule-based systems and large language models, offering a promising direction for future research in biomedical Natural Language Processing.

As we continue to refine these tools, future work will focus on expanding the existing datasets and addressing the discrepancies between gold-standard image labels and those extracted from radiology reports.

Code and Data Availability

Code for RadPert and RadPrompt is available on GitHub[5]. The CUH dataset is planned to be released in the following months while managed and made available through the hospital's clinical informatics unit.

Ethical Considerations

For the MIMIC-CXR gold-standard test set, access to LLMs through APIs conforms to the PhysioNet responsible use guidelines[6].

This ethical agreement with Cambridge University Hospitals currently limits the use of third-party APIs, but it is being revised prior to the dataset's release.

Acknowledgments

Panagiotis wishes to thank the David and Claudia Harding Foundation for their support through the University of Cambridge Harding Distinguished Postgraduate Scholars Programme. Anna Breger and Ian wish to acknowledge the EU/EFPIA Innovative Medicines Initiative 2 Joint Undertaking - DRAGON (101005122). Anna Breger is also supported by the Austrian Science Fund (FWF) through project P33217 and Ian by the National Institute for Health and Care Research (NIHR) Cambridge Biomedical Research Centre (BRC-1215-20014). Anna Korhonen wishes to acknowledge the UK Research and Innovation (UKRI) Frontier Research Grant (EP/Y031350/1).

[5] https://github.com/PanagiotisFytas/RadPert-RadPrompt.

[6] https://physionet.org/news/post/gpt-responsible-use

References

Lisa C Adams, Daniel Truhn, Felix Busch, Avan Kader, Stefan M Niehues, Marcus R Makowski, and Keno K Bressem. 2023. Leveraging gpt-4 for post hoc transformation of free-text radiology reports into structured reporting: a multilingual feasibility study. *Radiology*, 307(4):e230725.

Shruthi Bannur, Stephanie Hyland, Qianchu Liu, Fernando Perez-Garcia, Maximilian Ilse, Daniel C Castro, Benedikt Boecking, Harshita Sharma, Kenza Bouzid, Anja Thieme, et al. 2023. Learning to exploit temporal structure for biomedical vision-language processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15016–15027.

Benedikt Boecking, Naoto Usuyama, Shruthi Bannur, Daniel C Castro, Anton Schwaighofer, Stephanie Hyland, Maria Wetscherek, Tristan Naumann, Aditya Nori, Javier Alvarez-Valle, et al. 2022. Making the most of text semantics to improve biomedical vision–language processing. In *European conference on computer vision*, pages 1–21. Springer.

Selen Bozkurt, Emel Alkim, Imon Banerjee, and Daniel L Rubin. 2019. Automated detection of measurements and their descriptors in radiology reports using a hybrid natural language processing algorithm. *Journal of digital imaging*, 32:544–553.

Felix J Dorfner, Liv Jürgensen, Leonhard Donle, Fares Al Mohamad, Tobias R Bodenmann, Mason C Cleveland, Felix Busch, Lisa C Adams, James Sato, Thomas Schultz, et al. 2024. Is open-source there yet? a comparative study on commercial and open-source llms in their ability to label chest x-ray reports. *arXiv preprint arXiv:2402.12298*.

Bradley Efron and Robert Tibshirani. 1986. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical science*, pages 54–75.

Peter Floderus, Mirosław Kowaluk, Andrzej Lingas, and Eva-Marta Lundell. 2015. Induced subgraph isomorphism: Are some patterns substantially easier than others? *Theoretical Computer Science*, 605:119–128.

Jawook Gu, Han-Cheol Cho, Jiho Kim, Kihyun You, Eun Kyoung Hong, and Byungseok Roh. 2024. Chex-gpt: Harnessing large language models for enhanced chest x-ray report labeling. *arXiv preprint arXiv:2401.11505*.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23.

Saeed Hassanpour, Curtis P Langlotz, Timothy J Amrhein, Nicholas T Befera, and Matthew P Lungren. 2017. Performance of a machine learning classifier of knee mri reports in two large academic radiology practices: a tool to estimate diagnostic yield. *American Journal of Roentgenology*, 208(4):750–753.

Shih-Cheng Huang, Liyue Shen, Matthew P Lungren, and Serena Yeung. 2021. Gloria: A multimodal global-local representation learning framework for label-efficient medical image recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3942–3951.

A Infante, S Gaudino, F Orsini, A Del Ciello, C Gullì, B Merlino, L Natale, R Iezzi, and E Sala. 2024. Large language models (llms) in the evaluation of emergency radiology reports: performance of chatgpt-4, perplexity, and bard. *Clinical radiology*, 79(2):102–106.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn L. Ball, Katie S. Shpanskaya, Jayne Seekins, David A. Mong, Safwan S. Halabi, Jesse K. Sandberg, Ricky Jones, David B. Larson, Curtis P. Langlotz, Bhavik N. Patel, Matthew P. Lungren, and Andrew Y. Ng. 2019. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. *CoRR*, abs/1901.07031.

Saahil Jain, Ashwin Agrawal, Adriel Saporta, Steven Truong, Du Nguyen Duong Nguyen Duong, Tan Bui, Pierre Chambon, Yuhao Zhang, Matthew Lungren, Andrew Ng, Curtis Langlotz, Pranav Rajpurkar, and Pranav Rajpurkar. 2021a. Radgraph: Extracting clinical entities and relations from radiology reports. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Saahil Jain, Akshay Smit, Steven QH Truong, Chanh DT Nguyen, Minh-Thanh Huynh, Mudit Jain, Victoria A. Young, Andrew Y. Ng, Matthew P. Lungren, and Pranav Rajpurkar. 2021b. Visualchexbert: addressing the discrepancy between radiology report labels and image labels. In *Proceedings of the Conference on Health, Inference, and Learning*, CHIL '21, page 105–115, New York, NY, USA. Association for Computing Machinery.

Sanjay Jeganathan. 2023. The growing problem of radiologist shortages: Australia and new zealand's perspective. *Korean Journal of Radiology*, 24(11):1043.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317.

Sadhana Kalidindi and Sanjay Gandhi. 2023. Workforce crisis in radiology in the uk and the strategies to deal with it: Is artificial intelligence the saviour? *Cureus*, 15(8).

Kleanthis Konstantinidis. 2023. The shortage of radiographers: A global crisis in healthcare. *Journal of Medical Imaging and Radiation Sciences*.

Loïc Lannelongue, Jason Grealey, and Michael Inouye. 2021. Green algorithms: quantifying the carbon footprint of computation. *Advanced science*, 8(12):2100707.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2024. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. In *Proceedings of Machine Learning and Systems*, volume 6, pages 87–100.

Qianchu Liu, Stephanie Hyland, Shruthi Bannur, Kenza Bouzid, Daniel Castro, Maria Wetscherek, Robert Tinn, Harshita Sharma, Fernando Pérez-García, Anton Schwaighofer, Pranav Rajpurkar, Sameer Khanna, Hoifung Poon, Naoto Usuyama, Anja Thieme, Aditya Nori, Matthew Lungren, Ozan Oktay, and Javier Alvarez-Valle. 2023. Exploring the boundaries of GPT-4 in radiology. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14414–14445, Singapore. Association for Computational Linguistics.

OpenAI. 2023. Gpt-4 technical report.

Yifan Peng, Xiaosong Wang, Le Lu, Mohammadhadi Bagheri, Ronald Summers, and Zhiyong Lu. 2017. Negbio: a high-performance tool for negation and uncertainty detection in radiology reports. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2017:188–196.

Ewoud Pons, Loes MM Braun, MG Myriam Hunink, and Jan A Kors. 2016. Natural language processing in radiology: a systematic review. *Radiology*, 279(2):329–343.

Pranav Rajpurkar, Jeremy Irvin, Robyn L Ball, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis P Langlotz, et al. 2018. Deep learning for chest radiograph diagnosis: A retrospective comparison of the chexnext algorithm to practicing radiologists. *PLoS medicine*, 15(11):e1002686.

Pranav Rajpurkar, Chloe O'Connell, Amit Schechter, Nishit Asnani, Jason Li, Amirhossein Kiani, Robyn L Ball, Marc Mendelson, Gary Maartens, Daniël J van Hoving, et al. 2020. Chexaid: deep learning assistance for physician diagnosis of tuberculosis using chest x-rays in patients with hiv. *NPJ digital medicine*, 3(1):115.

Muhammad Imran Razzak, Saeeda Naz, and Ahmad Zaib. 2018. Deep learning for medical image processing: Overview, challenges and the future. *Classification in BioApps: Automation of decision making*, pages 323–350.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Eduardo P Reis, Joselisa PQ De Paiva, Maria CB Da Silva, Guilherme AS Ribeiro, Victor F Paiva, Lucas Bulgarelli, Henrique MH Lee, Paulo V Santos, Vanessa M Brito, Lucas TW Amaral, et al. 2022. Brax, brazilian labeled chest x-ray dataset. *Scientific Data*, 9(1):487.

Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Ng, and Matthew Lungren. 2020. Combining automatic labelers and expert annotations for accurate radiology report labeling using BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1500–1519, Online. Association for Computational Linguistics.

Yu-Xing Tang, You-Bao Tang, Yifan Peng, Ke Yan, Mohammadhadi Bagheri, Bernadette A Redd, Catherine J Brandon, Zhiyong Lu, Mei Han, Jing Xiao, et al. 2020. Automated abnormality classification of chest radiographs using deep convolutional neural networks. *NPJ digital medicine*, 3(1):70.

Ekin Tiu, Ellie Talius, Pujan Patel, Curtis P Langlotz, Andrew Y Ng, and Pranav Rajpurkar. 2022. Expert-level detection of pathologies from unannotated chest x-ray images via self-supervised learning. *Nature Biomedical Engineering*, 6(12):1399–1406.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5784–5789, Hong Kong, China. Association for Computational Linguistics.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M. Summers. 2017. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. *CoRR*, abs/1705.02315.

Yanshan Wang, Liwei Wang, Majid Rastegar-Mojarad, Sungrim Moon, Feichen Shen, Naveed Afzal, Sijia Liu, Yuqun Zeng, Saeed Mehrabi, Sunghwan Sohn, et al. 2018. Clinical information extraction applications: a literature review. *Journal of biomedical informatics*, 77:34–49.

Zifeng Wang, Zhenbang Wu, Dinesh Agarwal, and Jimeng Sun. 2022. MedCLIP: Contrastive learning from unpaired medical images and text. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3876–3887, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Alessandro Wollek, Sardi Hyska, Thomas Sedlmeyr, Philip Haitzer, Johannes Rueckel, Bastian O Sabel, Michael Ingrisch, and Tobias Lasser. 2024. German chexpert chest x-ray radiology report labeler. In *RöFo-Fortschritte auf dem Gebiet der Röntgenstrahlen und der bildgebenden Verfahren*. Georg Thieme Verlag KG.

David A Wood, Sina Kafiabadi, Aisha Al Busaidi, Emily Guilhem, Jeremy Lynch, Matthew Townend, Antanas Montvila, Juveria Siddiqui, Naveen Gadapa, Matthew Benger, et al. 2020. Labelling imaging datasets on the basis of neuroradiology reports: a validation study. In *Interpretable and Annotation-Efficient Learning for Medical Image Computing: Third International Workshop, iMIMIC 2020, Second International Workshop, MIL3ID 2020, and 5th International Workshop, LABELS 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4–8, 2020, Proceedings 3*, pages 254–265. Springer.

Appendix

	CheXpert	**RadPert**	**Llama-2 70B**	**RadPrompt /w Llama-2 70B**
Runtime (min)	7.1	4.8	41.8	43.6
CO$_2$e (g)	5.44	3.68	85.48	89.16
Device	CPU	CPU	GPU	GPU
Model	Core i7-6700k	Core i7-6700k	NVIDIA RTX 6000 Ada	NVIDIA RTX 6000 Ada

Table 5: Carbon footprint for inference on both MIMIC-CXR gold-standard and CUH test sets, as estimated utilizing the tools from Lannelongue et al. (2021). For RadPert, calculations include the extraction of the RadGraph knowledge graph. Notably, we are not able to provide estimates for GPT-4 Turbo, Gemini-1.5 Pro, and Claude-3 Sonnet since this information is not provided by the respective API providers.

	Negation Detection				**Uncertainty Detection**			
Pathologies	**F1 Score RadPert**		**Improvement over CheXpert (%)**		**F1 Score RadPert**		**Improvement over CheXpert (%)**	
Atelectasis	0.581	(0.000, 0.909)	61.6	(-41.8, 340.2)	0.386	(0.256, 0.511)	0.1	(-29.1, 44.7)
Cardiomegaly	0.834	(0.769, 0.892)	7.1	(0.6, 14.8)	0.093	(0.000, 0.227)	Inf.	(0.0, Inf.)
Consolidation	0.877	(0.762, 0.960)	-6.2	(-17.4, 2.8)	0.665	(0.488, 0.818)	269.7	(0.0, 909.7)
Edema	0.832	(0.773, 0.886)	8.7	(2.6, 16.5)	0.395	(0.160, 0.600)	104.2	(3.4, 275.3)
Enlarged Card.	0.916	(0.836, 0.982)	49.5	(21.9, 96.6)	0.062	(0.000, 0.207)	-3.3	(-28.6, 23.1)
Fracture	0.733	(0.444, 0.947)	0.0	(0.0, 0.0)	0.498	(0.000, 1.000)	Inf.	(0.0, Inf.)
Lung Lesion	0.422	(0.000, 0.800)	-5.1	(-50.0, 55.6)	0.128	(0.000, 0.400)	Inf.	(0.0, Inf.)
Lung Opacity	0.513	(0.353, 0.674)	32.2	(-17.3, 128.6)	0.000	(0.000, 0.000)	0.0	(0.0, 0.0)
Pler. Effusion	0.916	(0.871, 0.956)	-2.6	(-6.3, 1.3)	0.422	(0.267, 0.561)	-14.5	(-42.6, 22.8)
Pler. Other	0.000	(0.000, 0.000)	0.0	(0.0, 0.0)	0.000	(0.000, 0.000)	0.0	(0.0, 0.0)
Pneumonia	0.915	(0.867, 0.955)	17.3	(8.6, 29.0)	0.671	(0.582, 0.743)	43.8	(19.1, 76.5)
Pneumothorax	0.937	(0.912, 0.960)	2.1	(-0.7, 5.2)	0.645	(0.307, 0.909)	125.6	(-7.7, 540.1)
Sup. Devices	0.283	(0.000, 0.545)	0.0	(0.0, 0.0)	0.000	(0.000, 0.000)	0.0	(0.0, 0.0)
Macro Avg.	0.743	(0.686, 0.810)	4.1	(-4.9, 14.5)	0.453	(0.369, 0.554)	40.4	(9.5, 79.8)
Weighted Avg.	0.872	(0.852, 0.893)	5.9	(3.3, 8.7)	0.530	(0.460, 0.607)	31.4	(8.2, 61.3)
	Positive Mention Detection				**Mention Detection**			
Pathologies	**F1 Score RadPert**		**Improvement over CheXpert (%)**		**F1 Score RadPert**		**Improvement over CheXpert (%)**	
Atelectasis	0.819	(0.776, 0.859)	-5.8	(-10.2, -1.5)	0.944	(0.920, 0.965)	0.0	(0.0, 0.0)
Cardiomegaly	0.851	(0.806, 0.893)	7.5	(3.4, 12.0)	0.858	(0.826, 0.890)	-0.0	(-2.8, 3.0)
Consolidation	0.815	(0.724, 0.885)	8.6	(-0.6, 19.8)	0.930	(0.888, 0.963)	0.0	(0.0, 0.0)
Edema	0.809	(0.759, 0.859)	-8.2	(-12.9, -3.3)	0.887	(0.859, 0.916)	-0.2	(-0.9, 0.4)
Enlarged Card.	0.442	(0.336, 0.551)	1.3	(-21.3, 28.8)	0.529	(0.454, 0.609)	16.1	(-0.7, 36.7)
Fracture	0.902	(0.831, 0.964)	8.1	(0.9, 17.5)	0.952	(0.907, 0.990)	5.4	(-0.1, 12.4)
Lung Lesion	0.796	(0.702, 0.878)	1.9	(-6.4, 10.2)	0.834	(0.752, 0.901)	-2.4	(-7.0, 1.7)
Lung Opacity	0.819	(0.774, 0.859)	1.6	(-1.3, 4.5)	0.800	(0.757, 0.840)	-0.0	(-1.2, 1.2)
Pler. Effusion	0.889	(0.859, 0.916)	-3.1	(-6.3, 0.0)	0.979	(0.968, 0.989)	0.6	(-0.1, 1.4)
Pler. Other	0.592	(0.441, 0.727)	16.7	(1.6, 44.0)	0.592	(0.459, 0.709)	1.1	(-5.3, 11.3)
Pneumonia	0.654	(0.550, 0.744)	36.9	(8.5, 75.6)	0.952	(0.931, 0.971)	-0.5	(-1.5, 0.4)
Pneumothorax	0.765	(0.630, 0.870)	9.6	(-7.5, 30.4)	0.963	(0.945, 0.980)	-0.7	(-1.5, 0.0)
Sup. Devices	0.898	(0.869, 0.926)	1.3	(-0.7, 3.5)	0.893	(0.862, 0.918)	1.4	(-0.2, 3.1)
Macro Avg.	0.773	(0.749, 0.796)	4.1	(1.5, 6.8)	0.855	(0.839, 0.870)	1.0	(-0.0, 2.0)
Weighted Avg.	0.824	(0.809, 0.839)	0.9	(-0.8, 2.6)	0.899	(0.890, 0.908)	0.4	(-0.1, 0.9)

Table 6: F1 scores of RadPert and improvement over CheXpert on MIMIC-CXR gold-standard test set. We report results for the sub-tasks of negation detection, uncertainty detection, positive mention detection and mention detection. The scores correspond to the averages across 1000 bootstrap replicates and are reported alongside their confidence intervals.

	Weighted Average F1 Across Tasks			
	Gemini-1.5 Pro		Llama-2 70B	
Pathologies	**Base LLM**	**RadPrompt**	**Base LLM**	**RadPrompt**
Atelectasis	**0.838** **(0.792, 0.878)**	0.830 (0.787, 0.870)	**0.842** **(0.790, 0.884)**	0.782 (0.739, 0.822)
Cardiomegaly	**0.809** **(0.771, 0.842)**	0.790 (0.740, 0.835)	0.706 (0.657, 0.755)	**0.807** **(0.756, 0.853)**
Consolidation	0.588 (0.507, 0.662)	**0.743** **(0.665, 0.815)**	0.368 (0.302, 0.430)	**0.625** **(0.544, 0.700)**
Edema	0.737 (0.695, 0.778)	**0.794** **(0.752, 0.834)**	0.729 (0.686, 0.766)	**0.804** **(0.762, 0.845)**
Enlarged Card.	0.376 (0.275, 0.468)	**0.556** **(0.464, 0.643)**	0.248 (0.158, 0.343)	**0.619** **(0.537, 0.695)**
Fracture	0.718 (0.602, 0.820)	**0.792** **(0.696, 0.874)**	0.717 (0.583, 0.839)	**0.855** **(0.759, 0.932)**
Lung Lesion	0.413 (0.321, 0.508)	**0.678** **(0.575, 0.776)**	0.404 (0.313, 0.498)	**0.498** **(0.397, 0.595)**
Lung Opacity	0.587 (0.532, 0.638)	**0.744** **(0.700, 0.791)**	0.632 (0.583, 0.681)	**0.780** **(0.737, 0.824)**
Pleural Effusion	0.856 (0.829, 0.880)	**0.891** **(0.863, 0.916)**	0.827 (0.798, 0.853)	**0.867** **(0.837, 0.895)**
Pleural Other	0.409 (0.281, 0.535)	**0.492** **(0.346, 0.626)**	0.312 (0.129, 0.490)	**0.525** **(0.363, 0.669)**
Pneumonia	0.683 (0.635, 0.734)	**0.789** **(0.740, 0.836)**	**0.682** **(0.638, 0.724)**	0.645 (0.587, 0.698)
Pneumothorax	0.741 (0.699, 0.781)	**0.893** **(0.855, 0.926)**	0.730 (0.687, 0.773)	**0.871** **(0.825, 0.908)**
Support Devices	0.870 (0.836, 0.903)	**0.905** **(0.877, 0.932)**	0.718 (0.669, 0.767)	**0.883** **(0.851, 0.913)**
Macro Average	0.664 (0.642, 0.685)	**0.761** **(0.739, 0.783)**	0.609 (0.585, 0.631)	**0.736** **(0.714, 0.756)**
Weighted Average	0.740 (0.724, 0.755)	**0.815** **(0.799, 0.829)**	0.700 (0.684, 0.717)	**0.788** **(0.772, 0.803)**
	Claude-3 Sonnet		**GPT-4 Turbo**	
Pathologies	**Base LLM**	**RadPrompt**	**Base LLM**	**RadPrompt**
Atelectasis	**0.823** **(0.774, 0.868)**	0.812 (0.769, 0.850)	**0.864** **(0.819, 0.902)**	0.830 (0.785, 0.870)
Cardiomegaly	0.778 (0.742, 0.813)	**0.799** **(0.746, 0.845)**	**0.822** **(0.777, 0.858)**	0.806 (0.754, 0.849)
Consolidation	0.609 (0.530, 0.679)	**0.801** **(0.729, 0.865)**	0.804 (0.729, 0.864)	**0.825** **(0.752, 0.892)**
Edema	0.747 (0.702, 0.788)	**0.802** **(0.758, 0.846)**	**0.837** **(0.801, 0.875)**	0.811 (0.771, 0.853)
Enlarged Card.	0.289 (0.211, 0.369)	**0.578** **(0.494, 0.658)**	0.567 (0.474, 0.650)	**0.587** **(0.502, 0.667)**
Fracture	0.742 (0.622, 0.856)	**0.849** **(0.751, 0.929)**	0.785 (0.692, 0.868)	**0.826** **(0.732, 0.908)**
Lung Lesion	0.689 (0.586, 0.784)	**0.709** **(0.596, 0.808)**	0.634 (0.534, 0.725)	**0.675** **(0.565, 0.780)**
Lung Opacity	0.632 (0.574, 0.686)	**0.780** **(0.737, 0.823)**	0.724 (0.674, 0.769)	**0.782** **(0.738, 0.823)**
Pleural Effusion	0.832 (0.806, 0.858)	**0.901** **(0.875, 0.925)**	0.895 (0.869, 0.920)	**0.898** **(0.872, 0.923)**
Pleural Other	0.404 (0.278, 0.530)	**0.547** **(0.390, 0.692)**	0.558 (0.418, 0.680)	**0.616** **(0.462, 0.737)**
Pneumonia	0.640 (0.588, 0.688)	**0.780** **(0.727, 0.828)**	0.756 (0.699, 0.807)	**0.790** **(0.738, 0.838)**
Pneumothorax	0.684 (0.638, 0.723)	**0.922** **(0.887, 0.953)**	0.920 (0.890, 0.948)	**0.929** **(0.895, 0.960)**
Support Devices	0.886 (0.856, 0.914)	**0.896** **(0.867, 0.924)**	0.883 (0.849, 0.913)	**0.887** **(0.855, 0.918)**
Macro Average	0.674 (0.653, 0.693)	**0.783** **(0.761, 0.804)**	0.773 (0.752, 0.792)	**0.789** **(0.768, 0.808)**
Weighted Average	0.734 (0.718, 0.750)	**0.827** **(0.813, 0.841)**	0.824 (0.808, 0.838)	**0.832** **(0.818, 0.845)**

Table 7: Weighted F1 Scores across positive mention detection, negation detection, and uncertainty detection for RadPrompt on MIMIC-CXR gold-standard test set. The "Base LLM" column refers to the first-turn prediction of the LLM, and the "RadPrompt" column to the second-turn prediction. The scores correspond to the averages across 1000 bootstrap replicates and are reported alongside their confidence intervals.

	Mention Detection F1			
	Gemini-1.5 Pro		Llama-2 70B	
Pathologies	Base LLM	RadPrompt	Base LLM	RadPrompt
Atelectasis	0.785 (0.748, 0.819)	**0.926 (0.901, 0.949)**	0.746 (0.706, 0.785)	**0.939 (0.914, 0.961)**
Cardiomegaly	0.827 (0.793, 0.861)	**0.848 (0.815, 0.882)**	0.767 (0.731, 0.803)	**0.865 (0.832, 0.896)**
Consolidation	0.516 (0.450, 0.579)	**0.869 (0.817, 0.912)**	0.374 (0.318, 0.428)	**0.738 (0.672, 0.797)**
Edema	0.781 (0.745, 0.819)	**0.869 (0.838, 0.898)**	0.744 (0.704, 0.782)	**0.883 (0.854, 0.911)**
Enlarged Card.	0.409 (0.344, 0.480)	**0.504 (0.427, 0.582)**	0.326 (0.244, 0.414)	**0.534 (0.458, 0.614)**
Fracture	0.469 (0.385, 0.548)	**0.840 (0.767, 0.906)**	0.324 (0.258, 0.390)	**0.934 (0.881, 0.976)**
Lung Lesion	0.293 (0.236, 0.346)	**0.708 (0.625, 0.788)**	0.283 (0.229, 0.335)	**0.589 (0.495, 0.664)**
Lung Opacity	0.573 (0.526, 0.620)	**0.765 (0.724, 0.807)**	0.592 (0.545, 0.634)	**0.783 (0.743, 0.823)**
Pleural Effusion	0.913 (0.892, 0.932)	**0.966 (0.952, 0.978)**	0.883 (0.860, 0.907)	**0.964 (0.950, 0.977)**
Pleural Other	0.227 (0.158, 0.297)	**0.448 (0.323, 0.560)**	0.149 (0.091, 0.210)	**0.577 (0.444, 0.699)**
Pneumonia	0.802 (0.767, 0.838)	**0.940 (0.917, 0.960)**	0.714 (0.674, 0.753)	**0.890 (0.861, 0.915)**
Pneumothorax	0.760 (0.719, 0.797)	**0.941 (0.919, 0.961)**	0.758 (0.716, 0.795)	**0.943 (0.919, 0.964)**
Support Devices	0.804 (0.767, 0.837)	**0.888 (0.858, 0.913)**	0.655 (0.606, 0.701)	**0.892 (0.862, 0.917)**
Macro Average	0.627 (0.611, 0.647)	**0.809 (0.788, 0.829)**	0.563 (0.547, 0.580)	**0.810 (0.793, 0.827)**
Weighted Average	0.742 (0.724, 0.759)	**0.874 (0.861, 0.887)**	0.687 (0.670, 0.705)	**0.871 (0.860, 0.881)**
	Claude-3 Sonnet		GPT-4 Turbo	
Pathologies	Base LLM	RadPrompt	Base LLM	RadPrompt
Atelectasis	0.802 (0.767, 0.837)	**0.936 (0.911, 0.959)**	0.928 (0.901, 0.950)	**0.942 (0.918, 0.962)**
Cardiomegaly	0.777 (0.740, 0.813)	**0.858 (0.826, 0.890)**	0.858 (0.826, 0.889)	**0.859 (0.826, 0.892)**
Consolidation	0.50 (0.437, 0.561)	**0.921 (0.879, 0.956)**	0.882 (0.829, 0.922)	**0.930 (0.888, 0.963)**
Edema	0.789 (0.750, 0.826)	**0.884 (0.855, 0.911)**	**0.895 (0.868, 0.921)**	0.872 (0.842, 0.902)
Enlarged Card.	0.270 (0.222, 0.322)	**0.530 (0.453, 0.610)**	0.585 (0.505, 0.655)	**0.559 (0.478, 0.637)**
Fracture	0.442 (0.360, 0.522)	**0.933 (0.880, 0.976)**	0.811 (0.736, 0.883)	**0.885 (0.821, 0.942)**
Lung Lesion	0.398 (0.330, 0.464)	**0.847 (0.774, 0.908)**	0.701 (0.618, 0.776)	**0.799 (0.722, 0.868)**
Lung Opacity	0.564 (0.514, 0.612)	**0.790 (0.749, 0.830)**	0.742 (0.695, 0.786)	**0.795 (0.754, 0.834)**
Pleural Effusion	0.856 (0.830, 0.881)	**0.977 (0.966, 0.988)**	0.966 (0.953, 0.978)	**0.976 (0.965, 0.987)**
Pleural Other	0.211 (0.141, 0.278)	**0.592 (0.459, 0.709)**	0.560 (0.429, 0.674)	**0.630 (0.50, 0.744)**
Pneumonia	0.748 (0.708, 0.787)	**0.950 (0.928, 0.969)**	0.928 (0.905, 0.950)	**0.953 (0.932, 0.971)**
Pneumothorax	0.693 (0.651, 0.731)	**0.970 (0.953, 0.985)**	0.953 (0.934, 0.973)	**0.970 (0.953, 0.985)**
Support Devices	0.862 (0.831, 0.890)	**0.895 (0.866, 0.920)**	0.897 (0.868, 0.922)	**0.901 (0.875, 0.926)**
Macro Average	0.608 (0.591, 0.628)	**0.853 (0.837, 0.868)**	0.824 (0.804, 0.843)	**0.852 (0.836, 0.867)**
Weighted Average	0.720 (0.701, 0.739)	**0.897 (0.889, 0.906)**	0.881 (0.868, 0.892)	**0.897 (0.888, 0.907)**

Table 8: Mention detection F1 Scores for RadPrompt on MIMIC-CXR gold-standard test set. The "Base LLM" column refers to the first-turn prediction of the LLM, and the "RadPrompt" column to the second-turn prediction. The scores correspond to the averages across 1000 bootstrap replicates and are reported alongside their confidence intervals.

	Negation Detection F1							
	Gemini-1.5 Pro				Llama-2 70B			
Pathologies	**Base LLM**		**RadPrompt**		**Base LLM**		**RadPrompt**	
Atelectasis	0.066	(0.000, 0.143)	**0.340**	**(0.000, 0.616)**	0.015	(0.000, 0.047)	**0.579**	**(0.000, 0.909)**
Cardiomegaly	0.673	(0.594, 0.739)	**0.742**	**(0.667, 0.811)**	0.546	(0.477, 0.617)	**0.852**	**(0.789, 0.906)**
Consolidation	0.286	(0.188, 0.379)	**0.790**	**(0.654, 0.893)**	0.210	(0.136, 0.293)	**0.739**	**(0.591, 0.857)**
Edema	0.656	(0.583, 0.721)	**0.801**	**(0.737, 0.857)**	0.555	(0.483, 0.621)	**0.833**	**(0.769, 0.890)**
Enlarged Card.	0.455	(0.344, 0.561)	**0.696**	**(0.581, 0.804)**	0.125	(0.000, 0.294)	**0.887**	**(0.800, 0.962)**
Fracture	0.122	(0.043, 0.206)	**0.474**	**(0.235, 0.688)**	0.058	(0.020, 0.105)	**0.688**	**(0.400, 0.909)**
Lung Lesion	0.036	(0.000, 0.089)	**0.225**	**(0.000, 0.500)**	0.028	(0.000, 0.068)	**0.263**	**(0.000, 0.750)**
Lung Opacity	0.203	(0.101, 0.306)	**0.428**	**(0.256, 0.600)**	0.235	(0.142, 0.327)	**0.539**	**(0.373, 0.692)**
Pleural Effusion	0.733	(0.660, 0.798)	**0.888**	**(0.839, 0.932)**	0.625	(0.545, 0.698)	**0.870**	**(0.816, 0.923)**
Pleural Other	**0.035**	**(0.000, 0.087)**	0.000	(0.000, 0.000)	**0.023**	**(0.000, 0.057)**	0.000	(0.000, 0.000)
Pneumonia	0.624	(0.553, 0.692)	**0.887**	**(0.833, 0.933)**	0.498	(0.428, 0.567)	**0.823**	**(0.753, 0.888)**
Pneumothorax	0.714	(0.665, 0.756)	**0.922**	**(0.894, 0.948)**	0.710	(0.664, 0.753)	**0.909**	**(0.878, 0.937)**
Support Devices	0.059	(0.000, 0.143)	**0.207**	**(0.000, 0.414)**	0.026	(0.000, 0.065)	**0.295**	**(0.000, 0.556)**
Macro Average	0.371	(0.340, 0.416)	**0.628**	**(0.569, 0.693)**	0.302	(0.268, 0.350)	**0.717**	**(0.648, 0.787)**
Weighted Average	0.622	(0.590, 0.655)	**0.820**	**(0.788, 0.850)**	0.544	(0.511, 0.579)	**0.841**	**(0.818, 0.864)**
	Claude-3 Sonnet				GPT-4 Turbo			
Pathologies	**Base LLM**		**RadPrompt**		**Base LLM**		**RadPrompt**	
Atelectasis	0.099	(0.025, 0.198)	**0.581**	**(0.000, 0.909)**	0.515	(0.167, 0.800)	**0.868**	**(0.500, 1.000)**
Cardiomegaly	0.554	(0.478, 0.621)	**0.796**	**(0.721, 0.857)**	0.717	(0.640, 0.785)	**0.761**	**(0.684, 0.829)**
Consolidation	0.227	(0.151, 0.305)	**0.896**	**(0.788, 0.973)**	0.752	(0.615, 0.857)	**0.899**	**(0.800, 0.978)**
Edema	0.661	(0.589, 0.731)	**0.827**	**(0.768, 0.884)**	**0.871**	**(0.814, 0.921)**	0.836	(0.775, 0.893)
Enlarged Card.	0.148	(0.102, 0.199)	**0.713**	**(0.590, 0.818)**	0.620	(0.488, 0.736)	**0.741**	**(0.625, 0.841)**
Fracture	0.090	(0.031, 0.160)	**0.733**	**(0.444, 0.947)**	0.627	(0.333, 0.857)	**0.811**	**(0.545, 1.000)**
Lung Lesion	0.023	(0.000, 0.058)	**0.530**	**(0.000, 1.000)**	0.239	(0.000, 0.500)	**0.412**	**(0.000, 0.800)**
Lung Opacity	0.117	(0.059, 0.177)	**0.495**	**(0.326, 0.647)**	0.382	(0.217, 0.540)	**0.494**	**(0.318, 0.653)**
Pleural Effusion	0.572	(0.500, 0.644)	**0.937**	**(0.897, 0.973)**	0.903	(0.855, 0.946)	**0.948**	**(0.910, 0.981)**
Pleural Other	**0.032**	**(0.000, 0.076)**	0.000	(0.000, 0.000)	0.305	(0.000, 0.632)	**0.425**	**(0.000, 1.000)**
Pneumonia	0.537	(0.466, 0.604)	**0.909**	**(0.859, 0.951)**	0.862	(0.802, 0.910)	**0.914**	**(0.866, 0.954)**
Pneumothorax	0.638	(0.591, 0.683)	**0.947**	**(0.924, 0.969)**	0.937	(0.913, 0.959)	**0.955**	**(0.934, 0.973)**
Support Devices	0.174	(0.000, 0.350)	**0.259**	**(0.000, 0.500)**	**0.182**	**(0.000, 0.545)**	0.123	(0.000, 0.375)
Macro Average	0.305	(0.276, 0.340)	**0.731**	**(0.673, 0.795)**	0.640	(0.571, 0.715)	**0.757**	**(0.697, 0.832)**
Weighted Average	0.532	(0.495, 0.571)	**0.862**	**(0.842, 0.882)**	0.827	(0.796, 0.855)	**0.867**	**(0.847, 0.888)**

Table 9: Negation detection F1 Scores for RadPrompt on MIMIC-CXR gold-standard test set. The "Base LLM" column refers to the first-turn prediction of the LLM, and the "RadPrompt" column to the second-turn prediction. The scores correspond to the averages across 1000 bootstrap replicates and are reported alongside their confidence intervals.

	Uncertainty Detection F1							
	Gemini-1.5 Pro				Llama-2 70B			
Pathologies	**Base LLM**		**RadPrompt**		**Base LLM**		**RadPrompt**	
Atelectasis	0.301	(0.136, 0.464)	**0.376**	**(0.208, 0.536)**	0.364	(0.143, 0.560)	**0.386**	**(0.256, 0.515)**
Cardiomegaly	**0.385**	**(0.235, 0.529)**	0.170	(0.044, 0.320)	0.000	(0.000, 0.000)	**0.095**	**(0.000, 0.227)**
Consolidation	0.258	(0.138, 0.386)	**0.448**	**(0.250, 0.643)**	0.236	(0.133, 0.341)	**0.542**	**(0.367, 0.706)**
Edema	0.253	(0.082, 0.410)	**0.317**	**(0.087, 0.522)**	0.382	(0.154, 0.571)	**0.382**	**(0.148, 0.585)**
Enlarged Card.	0.000	(0.000, 0.000)	**0.045**	**(0.000, 0.150)**	0.000	(0.000, 0.000)	**0.068**	**(0.000, 0.229)**
Fracture	0.292	(0.000, 0.800)	**0.341**	**(0.000, 1.000)**	0.000	(0.000, 0.000)	**0.405**	**(0.000, 1.000)**
Lung Lesion	0.041	(0.000, 0.092)	**0.136**	**(0.000, 0.324)**	0.035	(0.000, 0.086)	**0.085**	**(0.000, 0.276)**
Lung Opacity	0.000	(0.000, 0.000)	**0.000**	**(0.000, 0.000)**	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)
Pleural Effusion	**0.483**	**(0.333, 0.619)**	0.466	(0.296, 0.606)	0.488	(0.308, 0.654)	**0.434**	**(0.276, 0.571)**
Pleural Other	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)
Pneumonia	0.705	(0.621, 0.776)	**0.710**	**(0.624, 0.781)**	**0.704**	**(0.614, 0.788)**	0.592	(0.497, 0.678)
Pneumothorax	**0.652**	**(0.353, 0.870)**	0.568	(0.222, 0.834)	0.475	(0.000, 0.800)	**0.566**	**(0.250, 0.846)**
Support Devices	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)
Macro Average	0.393	(0.332, 0.462)	0.393	(0.322, 0.473)	0.394	(0.316, 0.476)	**0.407**	**(0.329, 0.496)**
Weighted Average	0.498	(0.432, 0.560)	**0.512**	**(0.445, 0.577)**	**0.513**	**(0.439, 0.584)**	0.478	(0.414, 0.548)
	Claude-3 Sonnet				**GPT-4 Turbo**			
Pathologies	**Base LLM**		**RadPrompt**		**Base LLM**		**RadPrompt**	
Atelectasis	0.310	(0.095, 0.522)	**0.398**	**(0.254, 0.530)**	**0.406**	**(0.200, 0.583)**	0.337	(0.182, 0.491)
Cardiomegaly	**0.476**	**(0.300, 0.634)**	0.096	(0.000, 0.227)	**0.474**	**(0.293, 0.644)**	0.248	(0.074, 0.429)
Consolidation	0.454	(0.286, 0.607)	**0.634**	**(0.444, 0.783)**	**0.651**	**(0.455, 0.815)**	0.648	(0.461, 0.810)
Edema	0.202	(0.048, 0.344)	**0.395**	**(0.154, 0.606)**	**0.498**	**(0.222, 0.714)**	0.496	(0.222, 0.706)
Enlarged Card.	0.000	(0.000, 0.000)	**0.062**	**(0.000, 0.207)**	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)
Fracture	**0.722**	**(0.000, 1.000)**	0.498	(0.000, 1.000)	0.342	(0.000, 1.000)	**0.498**	**(0.000, 1.000)**
Lung Lesion	**0.275**	**(0.062, 0.500)**	0.120	(0.000, 0.375)	**0.287**	**(0.000, 0.526)**	0.186	(0.000, 0.545)
Lung Opacity	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)
Pleural Effusion	**0.583**	**(0.390, 0.735)**	0.490	(0.318, 0.633)	**0.469**	**(0.292, 0.625)**	0.464	(0.300, 0.615)
Pleural Other	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)
Pneumonia	**0.688**	**(0.595, 0.765)**	0.687	(0.598, 0.761)	0.683	(0.599, 0.758)	**0.708**	**(0.626, 0.780)**
Pneumothorax	**0.652**	**(0.363, 0.880)**	0.645	(0.307, 0.909)	**0.651**	**(0.308, 0.889)**	0.645	(0.307, 0.909)
Support Devices	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	**0.000**	**(0.000, 0.000)**
Macro Average	**0.493**	**(0.418, 0.567)**	0.460	(0.374, 0.569)	**0.521**	**(0.448, 0.599)**	0.511	(0.425, 0.592)
Weighted Average	**0.546**	**(0.484, 0.606)**	0.543	(0.469, 0.618)	**0.579**	**(0.516, 0.639)**	0.564	(0.497, 0.627)

Table 10: Uncertainty detection F1 Scores for RadPrompt on MIMIC-CXR gold-standard test set. The "Base LLM" column refers to the first-turn prediction of the LLM, and the "RadPrompt" column to the second-turn prediction. The scores correspond to the averages across 1000 bootstrap replicates and are reported alongside their confidence intervals.

	Positive Mention Detection F1							
	Gemini-1.5 Pro				Llama-2 70B			
Pathologies	**Base LLM**		**RadPrompt**		**Base LLM**		**RadPrompt**	
Atelectasis	**0.900**	**(0.869, 0.928)**	0.878	(0.843, 0.910)	**0.899**	**(0.866, 0.927)**	0.819	(0.775, 0.859)
Cardiomegaly	**0.928**	**(0.895, 0.957)**	0.879	(0.840, 0.914)	0.869	(0.828, 0.905)	**0.850**	**(0.805, 0.892)**
Consolidation	**0.811**	**(0.729, 0.881)**	0.810	(0.733, 0.883)	0.469	(0.382, 0.550)	**0.597**	**(0.496, 0.689)**
Edema	**0.824**	**(0.775, 0.867)**	0.822	(0.773, 0.867)	**0.866**	**(0.822, 0.906)**	0.816	(0.767, 0.866)
Enlarged Card.	0.327	(0.185, 0.472)	**0.468**	**(0.347, 0.583)**	0.336	(0.242, 0.437)	**0.446**	**(0.336, 0.557)**
Fracture	0.851	(0.762, 0.921)	**0.868**	**(0.787, 0.939)**	0.883	(0.800, 0.950)	**0.902**	**(0.831, 0.964)**
Lung Lesion	0.495	(0.409, 0.578)	**0.788**	**(0.701, 0.865)**	0.485	(0.396, 0.568)	**0.573**	**(0.479, 0.662)**
Lung Opacity	0.638	(0.584, 0.688)	**0.786**	**(0.743, 0.827)**	0.684	(0.636, 0.731)	**0.811**	**(0.769, 0.852)**
Pleural Effusion	0.917	(0.891, 0.939)	**0.918**	**(0.893, 0.941)**	**0.909**	**(0.883, 0.933)**	0.894	(0.864, 0.920)
Pleural Other	0.438	(0.312, 0.561)	**0.532**	**(0.387, 0.659)**	0.335	(0.136, 0.526)	**0.569**	**(0.393, 0.704)**
Pneumonia	0.723	(0.634, 0.806)	**0.744**	**(0.649, 0.821)**	**0.856**	**(0.792, 0.912)**	0.493	(0.398, 0.584)
Pneumothorax	**0.880**	**(0.792, 0.950)**	0.817	(0.704, 0.906)	**0.872**	**(0.781, 0.945)**	0.754	(0.625, 0.864)
Support Devices	0.887	(0.857, 0.915)	**0.920**	**(0.894, 0.945)**	0.733	(0.683, 0.780)	**0.896**	**(0.865, 0.922)**
Macro Average	0.740	(0.719, 0.762)	**0.787**	**(0.763, 0.809)**	0.708	(0.683, 0.731)	**0.725**	**(0.701, 0.749)**
Weighted Average	0.814	(0.800, 0.827)	**0.845**	**(0.831, 0.858)**	0.785	(0.770, 0.800)	**0.799**	**(0.784, 0.814)**
	Claude-3 Sonnet				GPT-4 Turbo			
Pathologies	**Base LLM**		**RadPrompt**		**Base LLM**		**RadPrompt**	
Atelectasis	**0.882**	**(0.849, 0.913)**	0.850	(0.810, 0.887)	**0.909**	**(0.876, 0.936)**	0.871	(0.833, 0.903)
Cardiomegaly	**0.937**	**(0.908, 0.964)**	0.870	(0.828, 0.907)	**0.915**	**(0.881, 0.947)**	0.887	(0.845, 0.923)
Consolidation	**0.811**	**(0.724, 0.880)**	0.808	(0.720, 0.882)	**0.868**	**(0.793, 0.930)**	0.844	(0.760, 0.914)
Edema	**0.841**	**(0.790, 0.885)**	0.813	(0.762, 0.861)	**0.839**	**(0.793, 0.883)**	0.816	(0.765, 0.864)
Enlarged Card.	0.389	(0.271, 0.504)	**0.493**	**(0.379, 0.603)**	**0.539**	**(0.415, 0.652)**	0.492	(0.378, 0.602)
Fracture	0.864	(0.775, 0.938)	**0.881**	**(0.805, 0.946)**	0.829	(0.742, 0.908)	**0.837**	**(0.750, 0.913)**
Lung Lesion	0.796	(0.716, 0.871)	**0.806**	**(0.727, 0.877)**	0.710	(0.621, 0.789)	**0.762**	**(0.673, 0.841)**
Lung Opacity	0.700	(0.644, 0.753)	**0.817**	**(0.773, 0.858)**	0.769	(0.717, 0.817)	**0.820**	**(0.775, 0.859)**
Pleural Effusion	**0.926**	**(0.902, 0.947)**	0.917	(0.892, 0.941)	**0.920**	**(0.896, 0.943)**	0.910	(0.885, 0.934)
Pleural Other	0.434	(0.309, 0.557)	**0.592**	**(0.441, 0.727)**	0.571	(0.419, 0.693)	**0.624**	**(0.480, 0.747)**
Pneumonia	0.706	(0.607, 0.791)	**0.713**	**(0.611, 0.800)**	0.698	(0.595, 0.786)	**0.720**	**(0.621, 0.803)**
Pneumothorax	**0.895**	**(0.812, 0.963)**	0.858	(0.767, 0.938)	**0.891**	**(0.800, 0.958)**	0.867	(0.766, 0.949)
Support Devices	0.901	(0.873, 0.927)	**0.910**	**(0.882, 0.936)**	0.898	(0.870, 0.924)	**0.904**	**(0.876, 0.928)**
Macro Average	0.776	(0.755, 0.795)	**0.795**	**(0.772, 0.816)**	**0.797**	**(0.773, 0.817)**	0.796	(0.773, 0.816)
Weighted Average	0.837	(0.823, 0.850)	**0.843**	**(0.828, 0.857)**	**0.849**	**(0.834, 0.863)**	0.846	(0.831, 0.860)

Table 11: Positive mention detection F1 Scores for RadPrompt on MIMIC-CXR gold-standard test set. The “Base LLM” column refers to the first-turn prediction of the LLM, and the “RadPrompt” column to the second-turn prediction. The scores correspond to the averages across 1000 bootstrap replicates and are reported alongside their confidence intervals.

Pathologies	Weighted F1		Improvement over		Improvement over	
	Llama-2 RadPrompt		1^{st} Turn Llama-2 (%)		RadPert (%)	
Atelectasis	0.830	(0.767, 0.888)	37.9	(22.9, 56.8)	-7.1	(-11.6, -3.0)
Cardiomegaly	0.810	(0.747, 0.867)	41.3	(28.6, 57.9)	-11.0	(-16.6, -6.0)
Consolidation	0.929	(0.903, 0.953)	27.7	(21.7, 34.8)	-2.3	(-4.2, -0.7)
Edema	0.529	(0.381, 0.639)	41.5	(-4.1, 99.8)	-15.1	(-27.3, -0.8)
Enlarged Card.	0.844	(0.790, 0.894)	Inf.	(Inf., Inf.)	-7.0	(-10.6, -3.3)
Fracture	0.684	(0.531, 0.817)	12.6	(-5.8, 38.8)	-10.3	(-20.0, -0.6)
Lung Lesion	0.699	(0.577, 0.817)	191.8	(132.3, 268.1)	-14.3	(-23.3, -5.7)
Lung Opacity	0.692	(0.636, 0.748)	2.9	(-6.5, 13.4)	-2.8	(-5.7, -0.1)
Pleur. Effusion	0.615	(0.562, 0.665)	-22.6	(-29.5, -16.1)	-3.9	(-7.6, -0.9)
Pleur. Other	0.106	(0.059, 0.163)	-80.9	(-89.5, -70.8)	34.2	(-8.5, 104.7)
Pneumonia	0.519	(0.374, 0.654)	259.1	(144.6, 433.0)	-21.0	(-33.4, -10.7)
Pneumothorax	0.606	(0.550, 0.661)	-16.0	(-23.4, -8.2)	-3.3	(-6.0, -0.2)
Sup. Devices	0.822	(0.785, 0.857)	6.2	(-0.3, 13.5)	-4.2	(-6.3, -2.3)
Macro Avg.	0.668	(0.639, 0.694)	27.4	(21.9, 32.6)	-8.0	(-10.2, -5.7)
Weighted Avg.	0.695	(0.668, 0.748)	3.0	(-1.3, 10.4)	-11.7	(-13.3, -5.3)

Table 12: Weighted average F1 scores for Llama-2-based RadPrompt on the CUH test set, alongside improvements over 1^{st} turn Llama-2 and RadPert predictions. The scores correspond to the averages across 1000 bootstrap replicates and are reported alongside their confidence intervals.

	Sub-Task F1 Scores							
	Negation Detection				**Uncertainty Detection**			
Pathologies	**Base Llama-2**		**RadPrompt**		**Base Llama-2**		**RadPrompt**	
Atelectasis	0.175	(0.111, 0.238)	**0.853**	**(0.766, 0.923)**	0.000	(0.000, 0.000)	**0.126**	**(0.000, 0.400)**
Cardiomegaly	0.579	(0.515, 0.639)	**0.835**	**(0.779, 0.884)**	0.000	(0.000, 0.000)	**0.412**	**(0.000, 0.727)**
Consolidation	0.665	(0.608, 0.720)	**0.923**	**(0.887, 0.953)**	0.145	(0.000, 0.298)	**0.490**	**(0.154, 0.769)**
Edema	0.160	(0.102, 0.223)	**0.444**	**(0.278, 0.597)**	0.322	(0.000, 1.000)	**0.408**	**(0.000, 0.800)**
Enlarged Card.	0.000	(0.000, 0.000)	**0.904**	**(0.854, 0.947)**	0.000	(0.000, 0.000)	**0.639**	**(0.471, 0.791)**
Fracture	0.052	(0.018, 0.098)	**0.269**	**(0.071, 0.483)**	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)
Lung Lesion	0.285	(0.215, 0.359)	**0.790**	**(0.684, 0.884)**	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)
Lung Opacity	0.022	(0.000, 0.056)	**0.187**	**(0.000, 0.421)**	**0.228**	**(0.000, 0.667)**	0.000	(0.000, 0.000)
Pleural Effusion	**0.758**	**(0.717, 0.797)**	0.532	(0.468, 0.592)	**0.414**	**(0.000, 0.800)**	0.319	(0.000, 0.600)
Pleural Other	**0.556**	**(0.490, 0.615)**	0.035	(0.000, 0.077)	0.000	(0.000, 0.000)	**0.515**	**(0.000, 1.000)**
Pneumonia	0.113	(0.063, 0.171)	**0.642**	**(0.424, 0.813)**	0.128	(0.028, 0.237)	**0.310**	**(0.087, 0.522)**
Pneumothorax	**0.730**	**(0.683, 0.771)**	0.610	(0.551, 0.663)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)
Support Devices	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)
Macro Average	0.377	(0.356, 0.413)	**0.596**	**(0.550, 0.664)**	0.296	(0.149, 0.441)	**0.459**	**(0.335, 0.585)**
Weighted Average	0.617	(0.588, 0.643)	**0.607**	**(0.568, 0.705)**	0.263	(0.127, 0.413)	**0.506**	**(0.387, 0.616)**
	Positive Mention Detection				**Mention Detection**			
Pathologies	**Base Llama-2**		**RadPrompt**		**Base Llama-2**		**RadPrompt**	
Atelectasis	**0.889**	**(0.826, 0.938)**	0.843	(0.779, 0.902)	0.454	(0.394, 0.510)	**0.868**	**(0.822, 0.908)**
Cardiomegaly	0.885	(0.750, 0.976)	**0.888**	**(0.765, 0.977)**	0.625	(0.567, 0.679)	**0.851**	**(0.805, 0.895)**
Consolidation	0.806	(0.761, 0.851)	**0.950**	**(0.924, 0.975)**	0.737	(0.704, 0.771)	**0.966**	**(0.951, 0.980)**
Edema	**0.828**	**(0.631, 0.960)**	0.697	(0.400, 0.917)	0.230	(0.167, 0.295)	**0.546**	**(0.405, 0.659)**
Enlarged Card.	0.000	(0.000, 0.000)	0.000	(0.000, 0.000)	0.026	(0.000, 0.065)	**0.876**	**(0.829, 0.919)**
Fracture	0.843	(0.711, 0.947)	**0.847**	**(0.722, 0.955)**	0.196	(0.136, 0.257)	**0.637**	**(0.500, 0.761)**
Lung Lesion	0.094	(0.021, 0.172)	**0.434**	**(0.000, 0.750)**	0.204	(0.154, 0.257)	**0.742**	**(0.635, 0.837)**
Lung Opacity	0.701	(0.655, 0.746)	**0.716**	**(0.659, 0.772)**	0.523	(0.476, 0.566)	**0.696**	**(0.638, 0.755)**
Pleural Effusion	**0.916**	**(0.875, 0.953)**	0.848	(0.789, 0.901)	**0.831**	**(0.801, 0.859)**	0.711	(0.665, 0.752)
Pleural Other	0.687	(0.451, 0.875)	**0.836**	**(0.640, 0.968)**	**0.577**	**(0.517, 0.635)**	0.162	(0.096, 0.239)
Pneumonia	0.187	(0.077, 0.298)	**0.479**	**(0.240, 0.684)**	0.147	(0.099, 0.193)	**0.580**	**(0.454, 0.693)**
Pneumothorax	**0.618**	**(0.333, 0.833)**	0.607	(0.286, 0.857)	**0.734**	**(0.691, 0.777)**	0.625	(0.568, 0.678)
Support Devices	0.780	(0.736, 0.819)	**0.828**	**(0.792, 0.863)**	0.646	(0.602, 0.688)	**0.818**	**(0.782, 0.852)**
Macro Average	0.687	(0.647, 0.723)	**0.752**	**(0.696, 0.798)**	0.461	(0.443, 0.499)	**0.698**	**(0.671, 0.723)**
Weighted Average	0.781	(0.763, 0.801)	**0.822**	**(0.799, 0.842)**	0.612	(0.590, 0.652)	**0.726**	**(0.704, 0.748)**

Table 13: F1 Scores for all sub-tasks for Llama-2-based RadPrompt on the CUH dataset. The “Base Llama-2” column refers to the first-turn prediction of the LLM, and the “RadPrompt” column to the second-turn prediction. The scores correspond to the averages across 1000 bootstrap replicates and are reported alongside their confidence intervals.

First Turn Prompt	**Second Turn Prompt**

Please accurately classify radiology reports for the presence or absence of findings. For each report, you will classify for the presence or absence of the following findings: Enlarged Cardiomediastinum, Cardiomegaly, ...
 	
Structure your answer like the template I provided to you delimited by triple backticks and return this template and nothing else.
 	
ALWAYS RETURN THE FULL TEMPLATE:
```
 {“Enlarged Cardiomediastinum”:
[ANSWER],
“Cardiomegaly”:
[ANSWER], ...
}
```
 	
If the existence of a finding is mentioned, answer “Yes”.
If a finding is mentioned as not existing, answer “No”.
If it cannot be determined if the patient has the findings, answer “Maybe”.
If a finding is not mentioned in the report, answer ‘Undefined”.
 	
Important steps to consider:
1. Read the radiology report and identify any mentions of Enlarged Cardiomediastinum, Cardiomegaly, ...
2. For every mention, determine if it is a positive, a negative, or an uncertain one.
3. If a finding is not mentioned in the report, answer “Undefined”.
4. For every finding, answer “Yes” if it is mentioned as existing (positive), “Maybe” if it is mentioned as uncertain, and “No” if it is mentioned as not existing (negative).
 	
Classify the following radiology report according to the template. Always output the full template, even if a finding is not mentioned.
 	
<START OF REPORT>
...
<END OF REPORT>
<ANSWER:>
	I am using a rule-based expert model to verify your answer. Here are some insights. However, those suggestions may be wrong. Please give me your new answer after either accepting or rejecting some or all of these suggestions:
	
	1. The tool agrees that the overall report should be classified as “Yes” for Pneumonia.
	2. In agreement with your previous answer, the tool detected no mentions of Enlarged Cardiomediastinum, Cardiomegaly,...
	3. The tool did not detect any explicit mentions for Lung Lesion and, thus, its suggested output is “Undefined” for Lung Lesion.
	4. The tool considers Atelectasis as “Maybe” because of the sentence “...”. However, you previously classified the overall report as “Yes” for Atelectasis.
	
	Please use the same template for your revised answer:
	```
	 {“Enlarged Cardiomediastinum”:
	[ANSWER],
	“Cardiomegaly”:
	[ANSWER], ...
	}
	```

Table 14: Example of RadPrompt first and second-turn prompts. The first-turn prompts are adapted from (Dorfner et al., 2024).

	MIMIC-CXR Gold-Standard				CUH			
Pathologies	**Null**	**Negative**	**Uncertain**	**Positive**	**Null**	**Negative**	**Uncertain**	**Positive**
Atelectasis	469	4	17	197	538	41	3	68
Cardiomegaly	452	82	14	139	523	100	10	17
Consolidation	592	23	17	55	355	138	6	151
Edema	460	85	10	132	614	23	2	11
Enlarged Card.	617	28	1	41	536	90	23	1
Fracture	637	8	2	40	623	8	0	19
Lung Lesion	621	4	8	54	607	34	2	7
Lung Opacity	493	23	0	171	471	7	1	171
Pleural Effusion	317	82	18	270	311	240	6	93
Pleural Other	660	2	0	25	476	158	2	14
Pneumonia	464	83	62	78	617	14	8	11
Pneumothorax	461	179	8	39	403	237	2	8
Support Devices	453	5	0	229	369	1	1	279
Total	6696	608	157	1470	6443	1091	66	850

Table 15: Number of output classes per pathology for the MIMIC-CXR gold-standard test set and CUH dataset.

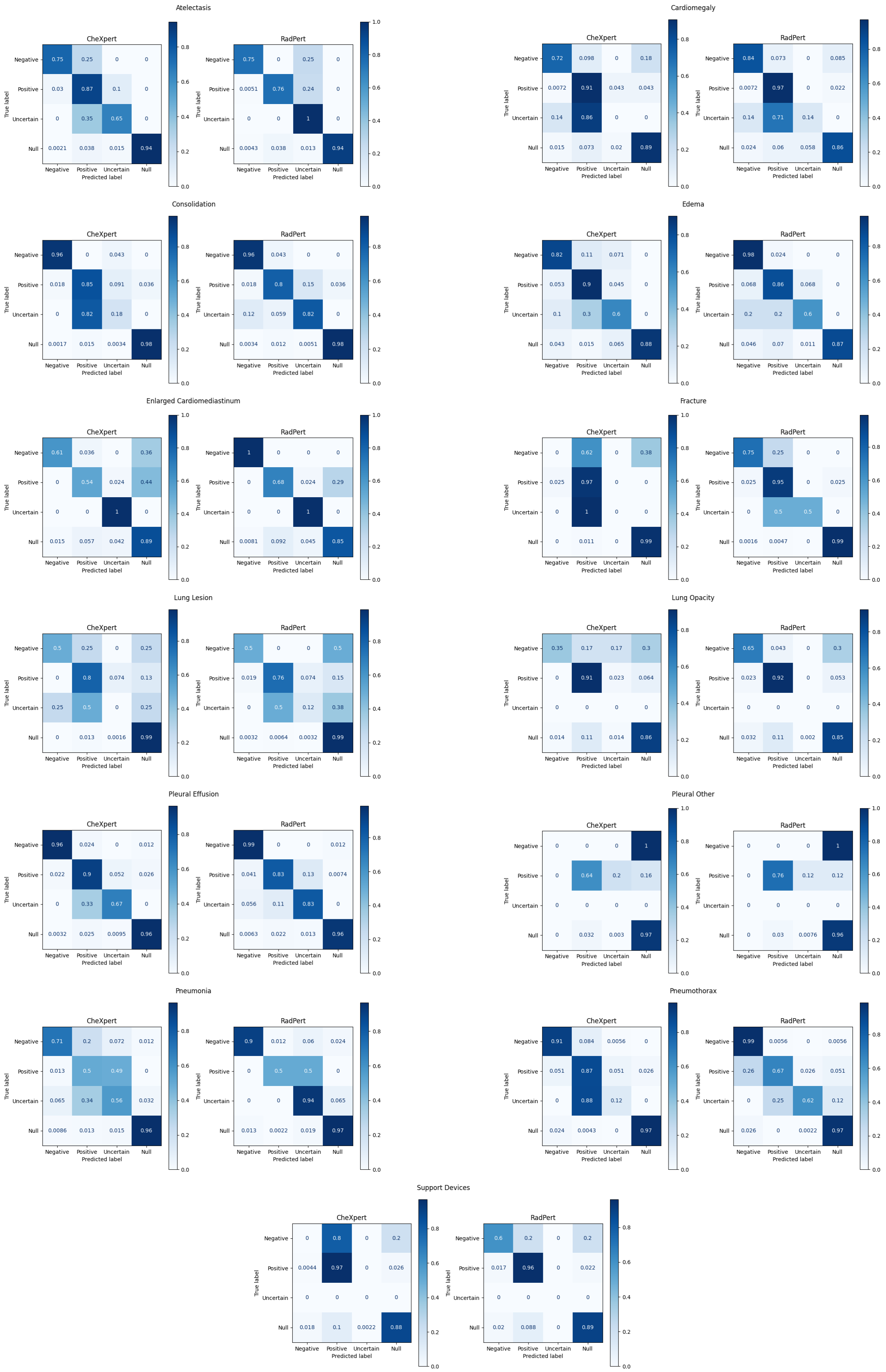


Figure 3: Normalized confusion matrices for MIMIC-CXR gold-standard test set.

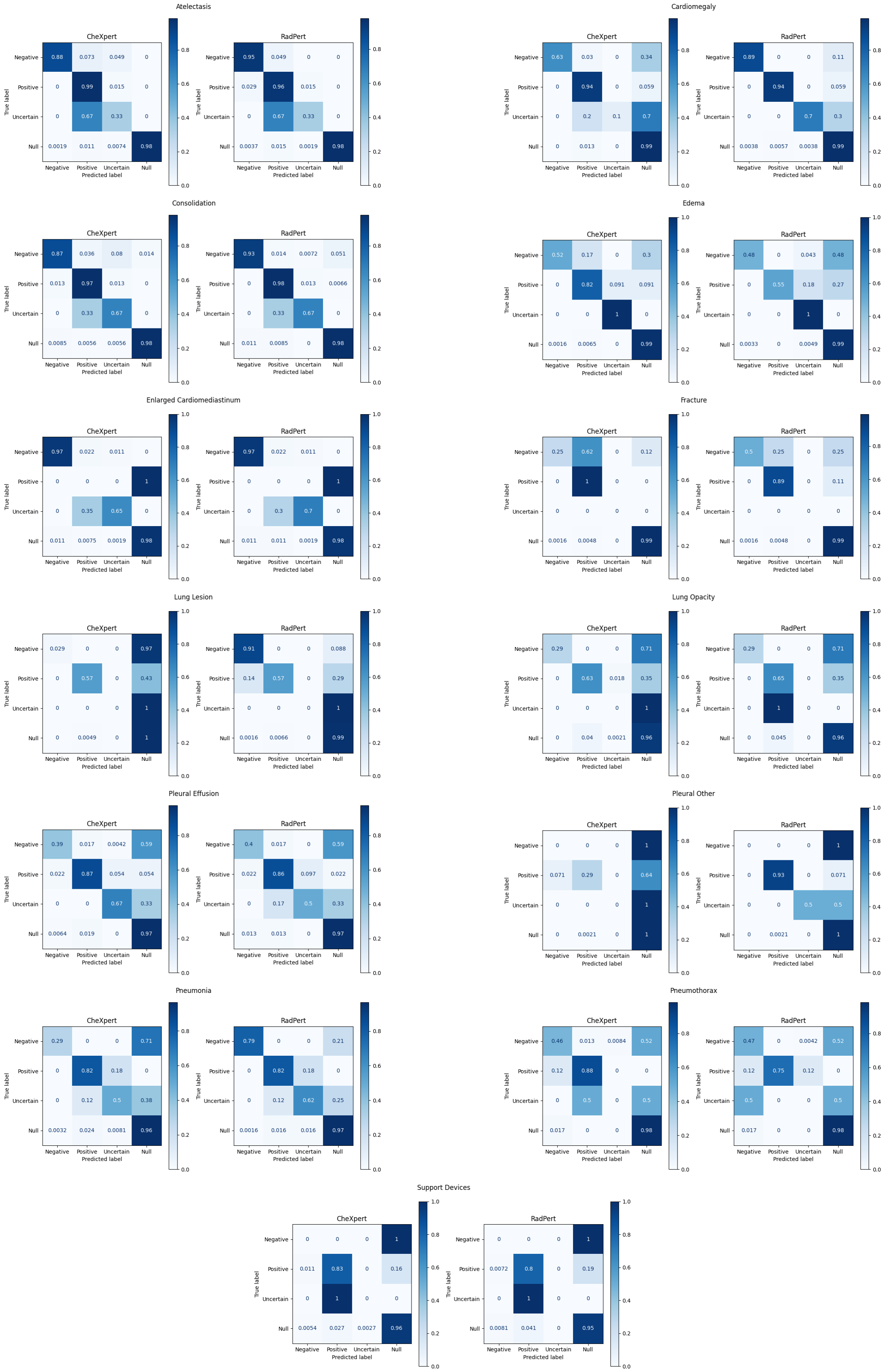


Figure 4: Normalized confusion matrices for CUH test set.

Using Large Language Models to Evaluate Biomedical Query-Focused Summarisation

Hashem Hijazi[1] and **Diego Mollá**[2,1] and **Vincent Nguyen**[1] and **Sarvnaz Karimi**[1]
[1]CSIRO Data61 and [2]Macquarie University
Sydney, Australia
firstname.lastname@csiro.au
Correspondence: diego.molla-aliod@mq.edu.au

Abstract

Biomedical question-answering systems remain popular for biomedical experts interacting with the literature to answer their medical questions. However, these systems are difficult to evaluate in the absence of costly human experts. Therefore, automatic evaluation metrics are often used in this space. Traditional automatic metrics such as ROUGE or BLEU, which rely on token overlap, have shown a low correlation with humans. We present a study that uses large language models (LLMs) to automatically evaluate systems from an international challenge on biomedical semantic indexing and question answering, called BioASQ. We measure the agreement of LLM-produced scores against human judgements. We show that LLMs correlate similarly to lexical methods when using basic prompting techniques. However, by aggregating evaluators with LLMs or by fine-tuning, we find that our methods outperform the baselines by a large margin, achieving a Spearman correlation of 0.501 and 0.511, respectively.

1 Introduction

Biomedical question answering (QA) is concerned with building systems that automatically answer biomedical questions posed by humans in natural language (Soares and Parreiras, 2018; Nguyen, 2019). To develop and optimise these systems, we must use metrics that evaluate the quality of their output. Automatic metrics such as ROUGE (Lin, 2004) and BLEU (Papineni et al., 2002) have been shown to correlate poorly with human evaluation (Liu et al., 2016), and human annotations are prohibitively expensive and impractical in the biomedical domain (Pampari et al., 2018; Guo et al., 2006). To rectify this problem, recent research has suggested using medium-sized model-based evaluators and Large Language Models (LLMs). Model-based evaluators such as BERTscore (Zhang et al., 2020) or BLEURT (Sellam et al., 2020) have demonstrated improvements over n-gram based metrics in various natural language generation (NLG) evaluation contexts such as summarisation and QA (Zhong et al., 2022). However, their evaluation capability is still far below that of humans.

The recent improvement of LLMs for various tasks has fostered research on their use for the evaluation of the performance of text generation tasks such as summarisation and dialogue generation (Liu et al., 2023). In this paper, we experiment with using LLMs to evaluate biomedical query-focused summarisation systems. To the best of our knowledge, this is the first study for such a task. In particular, we compare the correlation between human judgements and LLM-based evaluators for the evaluation of several systems participating in the "ideal answer" task of BioASQ 2021 and 2022 (Nentidis et al., 2022). Our study examines different prompting strategies for such evaluations.

2 Related Work

Fu et al. (2024) indicates that the use of LLMs as reference-free probability-based evaluators yields performance superior to n-gram metrics and model-based evaluators such as BERTscore, all the while providing a customised and multi-faceted evaluation with no training cost. Probability-based LLM evaluation, however, suffers from issues of robustness which lead to biases and loopholes (He et al., 2023) that impact its efficacy.

LLMs, through prompting, have also been used to evaluate text via Likert scale scoring (a five-level scale) (Likert, 1932), leading to greater performance than n-gram and model-based evaluation techniques. However, Chiang and Lee (2023) showed that outputting only a number could be sub-optimal but that asking the LLM to explain its rating can lead to an increase in correlation to human ratings. The pairwise ranking also showed promising results (Kotonya et al., 2023), with accuracy outperforming n-gram and model-based evaluators.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 236–242
August 16, 2024. ©2024 Association for Computational Linguistics

LLMs can also leverage emergent abilities which can be used to incorporate in-context learning (ICL) and chain-of-thought (CoT) in their evaluation strategies. In ICL (Xie et al., 2022), a model is provided with input-output examples of a downstream task instead of being trained or fine-tuned on the task. In CoT (Kojima et al., 2022), a complex task is broken down into multiple intermediate steps to improve reasoning in LLMs.

Liu et al. (2023) used GPT-4 to achieve the highest correlation with human evaluations in comparison to other model and n-gram based metrics. Jain et al. (2023), using GPT-3, showed that using few-shot prompting — a small number of examples added to the prompt — can reach or exceed the state-of-the-art on multi-dimensional evaluation and that this is robust to the sampling method of in-context examples whether it be random or representative of the range of scores in the example pool. However, Kotonya et al. (2023) showed using a small LLM (`orca-mini-v3-7B`), that one-shot prompting doesn't bring significantly greater results than zero-shot.

The use of LLMs as meta-evaluators who use their reasoning capabilities to combine diverse evaluation techniques has also seen promise. In Shu et al. (2023), various LLMs were given supplementary evaluation metrics like NLI Score (Bowman et al., 2015), BLEURT and probability-based LLM techniques to aid with their judgements, with the meta-evaluation outperforming all individual evaluators.

The above research, however, was not used in query-focused summarisation tasks. This paper is the first that tests the use of LLMs for the evaluation of biomedical query-focused summarisation.

3 Methodology

We use the human judgements of runs participating in the "ideal answers" question answering task of BioASQ (Nentidis et al., 2022). Such "ideal answers" are multi-sentence answers, and therefore the task is about biomedical query-focused summarisation. In particular, training and development is based on systems (runs) participating at BioASQ 2020[1], whereas testing is on runs participating at BioASQ 2021[2] and BioASQ 2022[3]. The rationale for using BioASQ 2020 for training is that it is the most recent year prior to the test data. Resource constraints do not allow us to use all runs participating at BioASQ 2020 for training, or all runs participating at BioASQ 2021 and 2022 for testing.

Evaluation criteria

Recall: *Fraction of information in the known answers that is reported in the generated response*
Precision: *Fraction of information in the generated response that is in the known answers*
Repetition: *Amount that the generated response repeats the same information*
Readability: *Generated response's ability to be easily understood and easily identifiable as an answer to the question by a human*

Figure 1: Human criteria for the evaluation of a (question, ideal answer) pair given the known answer. Each criterium was scored between 1 and 5.

The human judges are given instructions to evaluate the pair (question, generated answer), given a correct answer, according to four criteria presented in Figure 1. The final score of a (question, generated answer) pair is the average of the 4 criteria. These judgements are provided to us by the organisers of BioASQ. To preserve privacy, we only had access to the judgement of runs submitted by us.

For each automatic evaluation technique, the Spearman correlation (Spearman, 1904) is calculated. In addition, since the LLM-based evaluation generated integer numbers 1 to 5, for each LLM-based evaluation technique, the quadratic kappa, a well-established correlation metric for nominal scales (Cohen, 1968), was also calculated[4]. BioASQ runs five to six evaluation batches each year. Since there is no guarantee that the same runs participated in all batches, the correlations are computed separately for every batch, and the results reported in this paper are the average correlation. Each batch has approximately 100 (question, generated answer, known answer) triples.

Given that the automatic evaluation by LLMs can vary each time the LLM is run, each batch is evaluated three times and then the evaluation re-

[1] Run MQ1 in each batch submitted by Mollá et al. (2020).

[2] Run MQ1 in each batch submitted by Khanna and Mollá (2021).

[3] Run MQ1 in each batch submitted by Mollá (2022).

[4] Quadratic kappa could not be used on the output of the other evaluation techniques we tested because they generate real numbers between 0 and 1.

sults are averaged before computing the correlation with the human judges.

The LLM-based evaluations return independent evaluation scores for each evaluation criterion (Figure 1). Our correlation experiments consequently measure the correlation of each criterion (and average) per the corresponding human criterion (and average). For example, the *Precision* column of Table 1 shows the correlation of the *Precision* scores generated by each LLM evaluator for the *Precision* scores of the human judges.

4 Experiments

We investigated several evaluation strategies, from baseline well-known token-based metrics to the use of LLMs as evaluators in different settings.

4.1 Token-based Techniques and their Limitations

ROUGE-1 and ROUGE-2 (F1), as well as Sci-BERTscore (Precision, Recall, F1), were tested to attain a baseline correlation level. ROUGE F1 was chosen given its robustness over precision and recall (Mollá and Jones, 2020). Sci-BERT (Beltagy et al., 2019) is a BERT (Devlin et al., 2019) model pretrained on papers from Semantic Scholar, of which 82% are from the biomedical domain. Sci-BERT was chosen over BERT due to its greater understanding of biomedical terminology.

4.2 LLMs

GPT-3.5 (`gpt-3.5-turbo-0125`) and GPT-4 (`gpt-4-1106-preview`) were used as evaluators. All prompts included the question, reference answer(s), a system output, the defined evaluation criteria, and instructions rating responses on a 1-5 integer scale. For all runs, the system prompt was set to “You are a useful evaluator of a biomedical question answering system”, and the temperature and top p were set to 0 and 0.6, respectively, to facilitate reproducibility. Out-of-the-box (OTB) GPTs were tested with the base prompt listed in Figure 2.

4.2.1 Reason then Score

Chain-of-thought (CoT) prompting has demonstrated an increase in performance in various tasks, such as arithmetic and commonsense reasoning (Kojima et al., 2022). We used the variant of CoT called “Reason then Score” (RTS), in which an LLM is asked to explain its reasoning. RTS has emerged as a popular prompting technique (Shen et al., 2023). In our experiments, we altered the answer reporting format to include an explanation area that instructs the LLM to explain its answers (Figure 3).

OTB prompt

We have a biomedical question, a list of known answers, and the output generated by an automatic question-answering system. Given the known answers, evaluate the quality of the answer generated by the system. In your evaluation, address the following:
1. recall: The fraction of information in the known answers that is reported in the generated response. A higher score indicates better recall.
2. precision: The fraction of information in the generated response that is in the known answers. A higher score indicates better precision.
3. repetition: The amount that the generated response repeats the same information. A higher score indicates less repetition.
4. readability: The generated response's ability to be easily understood and easily identifiable as an answer to the question by a human. A higher score indicates better readability.
Use a 1-5 integer scale. Report the answer as a json structure using the template.
{"readability": {"score": 1-5}, "recall": {"score": 1-5}, "precision": {"score": 1-5}, "repetition": {"score": 1-5} }.
remember to report the answer as the format above with no deviations from this format. remember "score" is a key.

Figure 2: Prompt used in the out-of-the-box (OTB) LLM systems.

4.2.2 Few Shot

LLMs are reported to display an increase in performance when the prompt includes a few examples with their input query (Brown et al., 2020). We provided the LLMs with six examples from BioASQ 2020. Our initial experiments showed that random sampling of examples yielded poor performance. We instead used a percentile-based selection strategy to ensure a wide coverage of the example pool, based on the scores given by the human judges. In

RTS prompt

> ... (text of figure 2 inserted here, replacing its json template with the following) ...
> *{“recall”: {“explanation”:"", “score":1-5}*
> *“precision”: {“explanation”:"", “score":1-5}*
> *“repetition”: {“explanation”:"", “score":1-5}*
> *“readability”: {“explanation”:"", “score":1-5}*

Figure 3: Prompt used for *Reason then Score* (RTS) prompting.

particular, examples with 15th and 80th-percentile scores for recall, precision and readability were included in the prompt.

4.2.3 Fine-tuning

We also experimented with fine-tuned LLMs. In particular, we fine-tuned GPT-3.5 (`gpt-3.5-turbo`) using the same prompting format as OTB.

4.2.4 LLMs as Meta Evaluators

Inspired by work by Shu et al. (2023), we experimented with the use of LLMs as meta-evaluators of 3 different evaluators. In particular:

1. To aid the LLM in scoring repetition, we provided it with a “Repscore” which took the number of unique words in a response and divided it by the total number of words.
2. Smog score (Laughlin, 1969), which looks at the number of polysyllabic words and sentences in a text, was used to aid the scoring of readability.
3. Finally, the output from the fine-tuned GPT-3.5 model was also included to aid with the scoring of all dimensions.

Testing was done primarily on GPT-4, since GPT-3.5 showed a very limited capability in reasoning with other scores.

4.2.5 Pairwise Ranking

LLMs have also shown potential in comparative assessment (Liusie et al., 2024). We conducted pairwise ranking, where the LLM evaluator (gpt4-1106-preview) was given the output of two systems, the question, and was asked to rank them. We gave the LLM a CoT of the form Score then Reason as shown in Figure 4. We used accuracy and Cohen's kappa[5] to evaluate the performance.

Pairwise Ranking prompt

> We have a biomedical question, a list of known answers, and outputs generated by different automatic question answering systems.
> Given the known answers, rank the quality of the outputs generated by the system based on how similair they are to the ideal answers and if they answer the question properly.
> Report the answer as a JSON structure using the template, noting that a rank of 1 implies that this system output is the best compared to the others:
> {"ranks": 1: "system name", 2: "system name", "Explanation": "reasoning for why you ranked the systems this way. be specific"}.

Figure 4: Prompt used for *Pairwise ranking* prompting.

5 Results and Discussion

Table 1 shows the results of all experiments except pairwise ranking, and Table 2 shows the results of the experiments with pairwise ranking.

Similar to previous work, we find that token-based methods perform worse than LLMs in general (Liu et al., 2023). Possible causes of the relatively poor performance of token-based approaches is, as mentioned by Hanna and Bojar (2021), that ROUGE is unable to incorporate information on context and semantic meaning, whereas Sci-BERTscore is less sensitive to errors in text, especially if the candidate is lexically or stylistically similar, and both are insensitive to negation. These limitations have increasingly larger impacts on abstractive over extractive systems, making evaluating outputs using these metrics potentially unreliable. Still, Table 1 shows that the token-based methods achieve a correlation with the human judgements of precision and readability that is comparable to that of some of the LLM approaches.

[5] This is the standard Cohen's kappa, not quadratic kappa, since now the score is not a nominal scale.

Table 1: Spearman Correlation and Quadratic Kappa values. The Average column shows the average of Precision, Recall, Readability, and Repetition scores. The Combined column shows the correlation between the score resulting from averaging the machine predictions for Precision, Recall and averaging the human annotations. FS: Few Shot and CoT: Chain-of-Thought.

		Precision		Recall		Readability		Repetition		Average		Combined
		ρ	κ	ρ	κ	ρ	κ	ρ	κ	ρ	κ	ρ
Token-based	ROUGE-1-F1	0.384	-	0.280	-	0.159	-	0.118	-	0.235	-	0.357
	ROUGE-2-F1	0.412	-	0.271	-	0.164	-	0.129	-	0.244	-	0.364
	Sci-BERTscore-P	0.489	-	0.184	-	0.150	-	0.233	-	0.264	-	0.391
	Sci-BERTscore-R	0.283	-	0.325	-	0.146	-	0.152	-	0.227	-	0.341
	Sci-BERTscore-F1	0.420	-	0.264	-	0.154	-	0.208	-	0.262	-	0.391
LLM	GPT-3.5	0.370	0.235	0.298	0.229	0.123	0.090	0.232	0.143	0.256	0.174	0.388
	GPT-3.5 - CoT	0.322	0.266	0.293	0.256	0.168	0.151	0.242	0.155	0.256	0.207	0.376
	GPT-3.5 - FS	0.363	0.252	0.333	0.253	0.130	0.122	0.039	0.036	0.216	0.166	0.370
	Fine-tuned GPT-3.5	**0.537**	0.472	0.352	0.331	0.295	0.273	**0.516**	**0.460**	**0.425**	**0.384**	**0.511**
	GPT-4 as meta evaluator	0.531	**0.472**	**0.426**	**0.428**	**0.343**	**0.317**	0.388	0.275	0.422	0.373	0.501

Table 2: Accuracy and Kappa values of pairwise ranking evaluation.

	Accuracy	Kappa
Pairwise Ranking	0.61	0.31

GPT-3.5 with basic prompting displays similar performance to token-based metrics when using prompt engineering techniques such as CoT and few shot. When fine-tuned, GPT-3.5 attains a much higher correlation with humans than the token-based metrics. GPT-4 as a meta-evaluator achieves very similar results to the fine-tuned model. More testing is needed to be done on GPT-4 to see if prompt engineering leads to even better results.

Few-shot prompting performed the worst out of the LLM-based methods. In our preliminary experiments, we observed that the performance of the few-shot approach varied, with some batches increasing their correlations and others decreasing. This was in contrast with the performance of the fine-tuned approach, which had a lower variation across batches. This suggests that the examples chosen for few-shot could be more suited to certain batches. A promising future direction is to incorporate a dynamic selection of examples.

6 Conclusions

We compared the use of traditional evaluation metrics (ROUGE, BERTScore) with the use of LLMs for the evaluation of query-focused summarisation of biomedical questions. For this, we used system runs that participated in BioASQ challenge, and computed correlation between automatic evaluations and human judgements.

Our experiments show that, while LLMs with basic prompting do not outperform ROUGE or BERTScore, approaches that use fine-tuning or that combine LLMs with additional scorers, significantly improve correlation with human judgements.

7 Limitations

Due to limitations of resources and the availability of few runs, we have not experimented with a wide range of outputs of differing characteristics. Our training and test data used runs that were performed in the middle to the top range of systems participating in BioASQ. Therefore, the quality of the evaluators has not been tested on poor-quality runs. As a consequence, even though the results presented here should be valid to evaluate medium to high-quality systems, we cannot guarantee that the quality of the evaluations applies to poor-performing systems.

8 Ethical Considerations

The human judgements were obtained from the organisers of BioASQ. To ensure the privacy of these judgements, we only had access to judgements of past runs submitted by the authors of this paper, and the review judgements were anonymous.

Even though our results show a better correlation with human judgements than other automatic evaluation metrics, there is still room for improvement, and the evaluation results might not be reliable enough for applications requiring high-quality output systems and high-quality evaluation.

References

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: Pretrained language model for scientific text. In *EMNLP*.

Samuel Bowman, Gabor Angeli, Christopher Potts, and Christopher Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

Cheng-Han Chiang and Hung-yi Lee. 2023. A closer look into using large language models for automatic evaluation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8928–8942, Singapore.

Jacob Cohen. 1968. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychol Bull*, 70(4):213–20.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, Minneapolis, MN.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2024. GPTScore: Evaluate as you desire. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6556–6576, Mexico City, Mexico. Association for Computational Linguistics.

Yikun Guo, Robert Gaizauskas, Ian Roberts, and George Demetriou. 2006. Identifying personal health information using support vector machines. In *i2b2 Workshop on Challenges in Natural Language Processing for Clinical Data*, Washington, DC.

Michael Hanna and Ondřej Bojar. 2021. A fine-grained analysis of BERTScore. In *Proceedings of the Sixth Conference on Machine Translation*, pages 507–517, Online. Association for Computational Linguistics.

Tianxing He, Jingyu Zhang, Tianle Wang, Sachin Kumar, Kyunghyun Cho, James Glass, and Yulia Tsvetkov. 2023. On the blind spots of model-based evaluation metrics for text generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12067–12097, Toronto, Canada.

Sameer Jain, Vaishakh Keshava, Swarnashree Mysore Sathyendra, Patrick Fernandes, Pengfei Liu, Graham Neubig, and Chunting Zhou. 2023. Multi-dimensional evaluation of text summarization with in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2023*.

Urvashi Khanna and Diego Mollá. 2021. Transformer-based language models for factoid question answering at BioASQ9b. In *Proceedings of the Working Notes of CLEF 2021 – Conference and Labs of the Evaluation Forum*, pages 247–257.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *36th Conference on Neural Information Processing Systems (NeurIPS 2022)*.

Neema Kotonya, Saran Krishnasamy, Joel Tetreault, and Alejandro Jaimes. 2023. Little giants: Exploring the potential of small LLMs as evaluation metrics in summarization in the Eval4NLP 2023 shared task. In *Proceedings of the 4th Workshop on Evaluation and Comparison of NLP Systems*, pages 202–218, Bali, Indonesia.

G. Harry Mc Laughlin. 1969. Smog grading — a new readability formula. *Journal of Reading*, 12(8):639–646.

Rensis Likert. 1932. A technique for the measurement of attitudes. *Archives of Psychology*, 140:1–55.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the Association for Computational Linguistics Workshop*, pages 74–81, Barcelona, Spain.

Chia-Wei Liu, Ryan Lowe, Iulian Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132, Austin, Texas. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Adian Liusie, Potsawee Manakul, and Mark Gales. 2024. LLM comparative assessment: Zero-shot NLG evaluation through pairwise comparisons using large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 139–151, St. Julian's, Malta. Association for Computational Linguistics.

Diego Mollá. 2022. Query-focused extractive summarisation for biomedical and COVID-19 complex question answering. In *Proceedings of the Working Notes of CLEF 2022 – Conference and Labs of the Evaluation Forum*, pages 305–314.

Diego Mollá and Christopher Jones. 2020. Classification betters regression in query-based multi-document summarisation techniques for question answering. In *Machine Learning and Knowledge Discovery in Databases*, pages 624–635, Cham. Springer International Publishing.

Diego Mollá, Christopher Jones, and Vincent Nguyen. 2020. Query focused multi-document summarisation of biomedical texts. In *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum*.

Anastasios Nentidis, Georgios Katsimpras, Eirini Vandorou, Anastasia Krithara, Antonio Miranda-Escalada, Luis Gasco, Martin Krallinger, and Georgios Paliouras. 2022. Overview of BioASQ 2022: The tenth BioASQ challenge on large-scale biomedical semantic indexing and question answering. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 337–361, Cham. Springer International Publishing.

Vincent Nguyen. 2019. Question answering in the biomedical domain. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 54–63, Florence, Italy.

Anusri Pampari, Preethi Raghavan, Jennifer Liang, and Jian Peng. 2018. emrQA: A large corpus for question answering on electronic medical records. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2357–2368, Brussels, Belgium. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, US.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Chenhui Shen, Liying Cheng, Xuan-Phi Nguyen, Yang You, and Lidong Bing. 2023. Large language models are not yet human-level evaluators for abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4215–4233, Singapore.

Lei Shu, Nevan Wichers, Liangchen Luo, Yun Zhu, Yinxiao Liu, Jindong Chen, and Lei Meng. 2023. Fusion-Eval: Integrating evaluators with LLMs. *Preprint*, arXiv:2311.09204.

Marco Soares and Fernando Parreiras. 2018. A literature review on question answering techniques, paradigms and systems. *Journal of King Saud University - Computer and Information Sciences*.

C. Spearman. 1904. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR*.

Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022. Towards a unified multi-dimensional evaluator for text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2023–2038, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Continuous Predictive Modeling of Clinical Notes and ICD Codes in Patient Health Records

Mireia Hernandez Caralt **Clarence Boon Liang Ng** **Marek Rei**
Imperial College London, United Kingdom
{mireia.hernandez-caralt22,clarence.ng21,marek.rei}@imperial.ac.uk

Abstract

Electronic Health Records (EHR) serve as a valuable source of patient information, offering insights into medical histories, treatments, and outcomes. Previous research has developed systems for detecting applicable ICD codes that should be assigned while writing a given EHR document, mainly focusing on discharge summaries written at the end of a hospital stay. In this work, we investigate the potential of predicting these codes for the whole patient stay at different time points during their stay, even before they are officially assigned by clinicians. The development of methods to predict diagnoses and treatments earlier in advance could open opportunities for predictive medicine, such as identifying disease risks sooner, suggesting treatments, and optimizing resource allocation. Our experiments show that predictions regarding final ICD codes can be made already two days after admission and we propose a custom model that improves performance on this early prediction task.

1 Introduction

Electronic health records (EHR) are rich repositories of patient information, chronicling their medical history, diagnoses, treatment plans, medications and outcomes (Jensen et al., 2012; Johnson et al., 2016). The aggregation and modeling of this data over time presents a unique opportunity for revealing patterns that can improve patient care, operational efficiency, and healthcare delivery. Contained within the EHR are textual notes written by clinicians during patient encounters, which are essential for a comprehensive understanding of patient health. These free-text narratives stand out as a particularly rich source of nuanced information, but their unstructured format and domain-specific language use have left them largely underutilized, compared to more readily available structured data sources (Tayefi et al., 2021).

Category	**# notes**	**# words**
Discharge summ.	59,652	79,649,691
ECG	209,051	5,625,393
Echo	45,794	12,810,062
Nursing	1,046,053	185,856,841
Physician	141,624	322,961,183
Radiology	522,279	165,805,982
Respiratory	31,739	11,957,187
Other	26,988	13,086,023

Table 1: The number of clinical notes from different categories, along with the number of words in those notes, in the MIMIC-III dataset.

Health records are often accompanied by the International Classification of Diseases (ICD) codes – standardized codes that categorize diagnoses and procedures performed during clinical encounters (Cartwright, 2013). Assigning ICD codes manually is a highly time-consuming task necessary for billing, therefore previous research has been developing multi-label classification systems to detect applicable codes that should be assigned while writing a given document (Mullenbach et al., 2018; Liu et al., 2021). The research focus has been on classifying discharge summaries, which are written at the end of a patient's hospital stay (Ji et al., 2021; Dai et al., 2022). While this setup provides a useful proxy task, the complete EHR sequence is much longer, containing detailed reports from nursing and radiology, along with specialized notes on echographies and cardiograms (FCG), among others (Table 1). Recent work has argued that for most practical applications such code classification should be performed on earlier medical notes instead of the discharge summary (Cheng et al., 2023).

In this work, we investigate the potential of predicting ICD codes for the whole patient stay at different time points during their stay. Beyond the

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 243–255
August 16, 2024. ©2024 Association for Computational Linguistics

task of detecting codes for a given note, we treat ICD codes as a structured summary of all the treatments provided and diagnoses assigned during a hospital stay. The development of models for predicting this information early in the clinical timeline, based on partial indicators even before these codes have been officially assigned, would open many possibilities for predictive medicine. Such systems would go beyond the post-discharge diagnostic practices and could be used for identifying early disease risks, suggesting potential treatments or optimizing hospital resource allocation.

We investigate the feasibility of this novel task and evaluate to what extent the final set of ICD codes can be predicted at earlier stages during the hospital stay. In addition, we propose a custom model for this task that is able to improve prediction accuracy at different time steps. Unlike previous ICD code prediction models, the architecture is designed with causal attention to ensure that representations at any point throughout a patient's hospital stay are constructed based on the notes available up to that point, without accessing information in the future. The model is then optimized to predict ICD codes after every additional note in the input sequence, instead of only at the discharge summary, teaching it to make predictions at any chosen time point during the hospital stay. This task poses additional challenges, as the length of the complete EHR sequence far exceeds that of the discharge summary and early notes have a weaker correlation with the final labels. We introduce a novel method that both augments the data during training and extends the context during inference, substantially enhancing the performance on early ICD code prediction. The code for the model and the experiments are available online.[1]

2 Related Work

The closest previous research to ours has been on automating ICD code assignment. Given a document mentioning diagnoses and treatments in free-form text, the aim is to detect the correct codes that should be assigned by the clinician. The first attempts at this task primarily relied on convolutional neural networks (CNNs) (Mullenbach et al., 2018; Li and Yu, 2020; Liu et al., 2021) and long short-term memory networks (LSTMs) (Vu et al., 2020; Yuan et al., 2022). These models utilized pre-trained word2vec embeddings (Mikolov et al., 2013) and combined neighbouring word representations using convolutional filters or recurrent architectures. Despite their simplicity, some of these models achieved very high-performance baselines that were difficult to surpass with transformer approaches (Ji et al., 2021).

Efforts to apply pre-trained transformers without further modifications to the ICD coding problem were unsuccessful (Ji et al., 2021; Dai et al., 2022). The discharge summary contains 3,594 tokens on average, while the combined set of notes contains an average of 21,916 tokens per patient stay (Ng et al., 2023). Crucial information to predict patient diagnoses is likely to be dispersed throughout these notes, thus models with limited context length risk overlooking a significant portion of relevant data. For this reason, subsequent studies focused on adapting transformer architectures to process longer textual sequences.

PubMedBERT-hier (Ji et al., 2021) employed hierarchical transformers to mitigate the length limitation issue, obtaining substantially better results. This approach segments the document into chunks of 512 tokens and employs a BERT-based model pre-trained on the biomedical domain to encode each segment (Gu et al., 2022). The segments are then combined using a hierarchical transformer running over the CLS-token embeddings. The TrLDC model (Dai et al., 2022) further improved performance by employing a RoBERTa-based model pretrained from scratch on biomedical articles and clinical notes (Lewis et al., 2020).

The PAAT (Partition Attention) model (Kim, 2022) was able to surpass LSTM-based models like MSMN (Yuan et al., 2022) on the task of identifying the top 50 labels. PAAT combines the Clinical Long-Former and a bi-LSTM, employing partition-based label attention for improved performance. HiLAT (Hierarchical Label-Attention) (Liu et al., 2022) achieved strong results on the top 50 labels by utilizing ClinicalPlusXLNet, which outperforms other transformers like RoBERTa variants, with the downside being that the training speed is four times slower due to its bidirectional context capturing (Liu et al., 2022).

The HTDS (Hierarchical Transformer for Document Sequences) model (Ng et al., 2023) integrated earlier notes into the input context when making decisions about the discharge summary. This model employs a RoBERTa base transformer and a separate transformer layer running over the individual

[1] https://github.com/mireiahernandez/icd-continuous-prediction

token representations, not only the CLS embeddings. They found that the earlier notes were indeed useful as additional evidence at the end of the hospital stay and provided performance improvements when classifying discharge summaries.

All this prior work has trained systems to assign ICD codes at the end of the hospital stay, whereas we investigate models for making predictions at any point during the stay. Furthermore, while previous work has focused on detecting explicit mentions of diagnoses and treatments in a given text, we investigate to what extent future labels can be inferred based on only earlier documents.

3 Architecture

We investigate a model architecture that can be trained to encode a long temporal sequence of many clinical notes and make predictions at any time point, only using earlier notes as context. The model breaks the sequence into smaller chunks and encodes them using a hierarchical transformer. These chunks are combined with causal label attention, which gathers evidence with label-specific attention heads while ensuring that representations at any time are constructed based on the notes available up to that point, without accessing information in the future. Finally, a probability distribution across the labels is predicted at each possible time point. We refer to the model as a Label-Attentive Hierarchical Sequence Transformer (LAHST) and describe it in more detail below. Figure 1 provides a diagram of the architecture.

Step 1. Document splitting. Each document within the EHR sequence of a patient is tokenized and split into chunks of T tokens. Each patient has a variable total number of chunks, and during training, a maximum of N chunks is selected based on the criteria described in the next section.

Step 2. Chunk encoding. Each of the chunks is encoded with a pre-trained language model (PLM), extracting the CLS-token embedding as the representation, yielding a tensor $e \in \mathbb{R}^{N \times D}$. We use the `RoBERTa-base-PM-M3-Voc` checkpoint, as it has been trained on two domains that match our task closely: 1) PubMed and PMC, which cover biomedical publications, and 2) MIMIC-III, which contains clinical health records (Lewis et al., 2020).

Step 3. Causal attention. We augment a transformer layer with causal attention (Choromanski et al., 2021) in order to combine temporal information from any previous step without providing access to information in the future steps. At the same time, the whole sequence can be efficiently processed in parallel by masking any attention connections on the right side of the target position. This component takes as input the sequence of chunk embeddings $e \in \mathbb{R}^{N \times D}$ and generates a sequence of embeddings $h \in \mathbb{R}^{N \times D}$ which combines information over past documents:

$$h_i = CausalAttn(e_1, ..., e_i), i \in [1, N] \quad (1)$$

Step 4. Masked multi-head label attention. We apply label-wise attention (Mullenbach et al., 2018) with two key modifications: the use of multiple attention heads, and the use of causal masking to obtain temporal label-wise document embeddings. For each temporal position t, we define an attention mask $a_t \in \mathbb{R}^{L \times N}$ to prevent attention to future notes, which is constant in the label dimension, and nullifies attention weights beyond temporal position t.

$$a_t[:, i] = \begin{cases} 0, & \text{if } i \leq t \\ -\infty & \text{otherwise} \end{cases} \quad (2)$$

We then combine this mask with multi-head attention (Vaswani et al., 2017) using learnable label embeddings $q \in \mathbb{R}^{L \times D}$ as queries and the previously generated past context embeddings $h \in \mathbb{R}^{N \times D}$ as keys and values:

$$\begin{aligned} d_t &= \text{MultiHeadAttn}(q, h, h, a_t) \\ &= \text{Concat}(\text{head}_1, \cdots, \text{head}_H) W^o \end{aligned} \quad (3)$$

Here, each head inputs a linear projection of the key, query and value embeddings $e_{k,i} = W_i^K q$, $e_{q,i} = W_i^Q h$, $e_{v,i} = W_i^V h$ and applies masked attention (Choromanski et al., 2021) as follows:

$$\begin{aligned} \text{head}_i &= \text{Attention}(e_{k,i}, e_{q,i}, e_{v,i}, \text{mask} = a_t) \\ &= \text{SoftMax}(\frac{e_{k,i} e_{q,i}^T}{\sqrt{D/H}} + a_t) e_{v,i} \end{aligned} \quad (4)$$

This yields a sequence of label-wise document embeddings $d_t \in \mathbb{R}^{L \times D}$ for each position $t \in \{1, ..., N\}$. In practice, we obtain all the embeddings $d \in \mathbb{R}^{N \times L \times D}$ efficiently in one pass by assigning the batch dimension to the temporal dimension.

Step 5. Temporal label-wise predictions. Finally, temporal probabilities are calculated by projecting the embedding using linear weights $w \in$

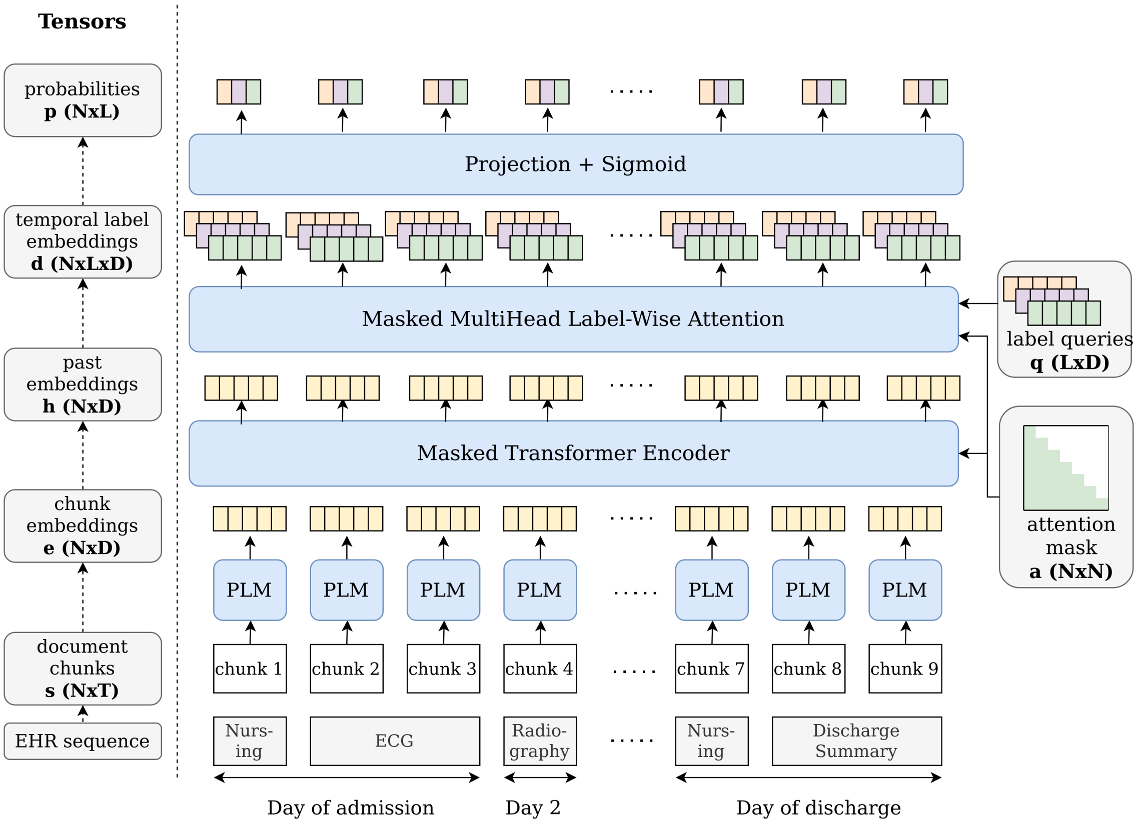


Figure 1: **LAHST** (**L**abel-**A**ttentive **H**ierarchical **S**equence **T**ransformer) architecture. Clinical notes generated throughout the hospital stay are split into chunks. Each chunk is encoded using a pre-trained language model (PLM) to extract the CLS-token embedding. Next, a hierarchical transformer encoder is applied, utilizing causal masking to combine information among past segment embeddings. Finally, the network generates a distinct document representation for each label and temporal point combination and these are then transformed into probabilities by the output layer.

$\mathbb{R}^{L\times D}$ followed by a sigmoid activation. The probability at time t for label l is calculated using the the label weight $w_l \in \mathbb{R}^D$ and the label document embedding at position t, denoted as $d_{t,l} \in \mathbb{R}^D$:

$$p_{t,l} = \text{sigmoid}(w_l \cdot d_{t,l}) \tag{5}$$

The output of the model is a probability matrix $p \in \mathbb{R}^{N\times L}$, containing probabilities for each label at each temporal point. The masking process within the transformer and label attention modules ensures that time t probability calculations consider only past documents. The model is trained using the binary cross-entropy loss.

4 Extending the Context

Hierarchical transformer architectures break long inputs into smaller components and reduce the number of long-distance attention operations, thereby keeping memory and computation requirements more manageable when processing very long sequences. This makes them well-suited for ICD code classification, as local context is more important for this task and hierarchical models have been shown to outperform long-context models in this setting (Dai et al., 2022). However, even hierarchical models have difficulty with very long sequences, particularly during training. The gradient must be backpropagated through each individual chunk encoding, which can easily cause memory issues when the models are large and the number of segments exceeds a maximum limit.

For this reason, we propose a novel solution for applying hierarchical transformers to very long document sequences, such as the sequences of notes in EHR. We refer to this method as the Extended Context Algorithm (ECA). It consists of the following modifications to the training and inference loops.

Training (Algorithm 1). For each episode of training, the loop iterates over the training dataset $\mathcal{D}_{train}$, processing each data sample (s, y), where s is the input sequence and y is the corresponding label. Within the loop, a random selection of

Algorithm 1 ECA Training loop

```
$\mathcal{D}_{train} \leftarrow$ training set (sequence-label pairs)
$N_{max} \leftarrow$ max. number of chunks
for each episode do
    for each $(s, y)$ in $\mathcal{D}_{train}$ do
        $m \leftarrow min(N_{max}, len(s))$
        select $m$ random indices $i_1, ...i_m$
        sort $i_1, \cdots, i_m$ in ascending order
        $s' \leftarrow [s[i_1], \cdots, s[i_m]]$
        $p, h \leftarrow$ model.forward$(s')$
        $\mathcal{L} \leftarrow BCE(y, p)$
        do backward pass and optimizer step
    end for
end for
```

Algorithm 2 ECA Inference loop

```
$\mathcal{D}_{test} \leftarrow$ test set (no labels)
$N_{max} \leftarrow$ max. number of chunks
for each $s$ in $\mathcal{D}_{test}$ do
    $h_{list} \leftarrow$empty list
    for $i$ in range$(0, len(s), N_{max})$ do
        $s_{batch} \leftarrow s[i : i + N_{max}]$
        $p_{batch}, h_{batch} \leftarrow$ model.forward$(s_{batch})$
        append $h_{batch}$ to $h_{list}$
    end for
    $h \leftarrow$ concatenate $h_{list}$ along batch dim.
    $p \leftarrow$ model.label_attention$(h)$
end for
```

notes is chosen to create a subset of the input sequence, with the maximum number of chunks set as N_{max}. These sub-sequences s' are then used for optimizing the model, each time sampling a slightly different training instance. Instead of trying to fit the whole sequence into the input during training, we sample notes and form multiple different shorter versions of the sequence for training the model. This has the added benefit of creating a data augmentation effect, as the model learns to make decisions based on different versions of the same datapoint.

Inference (Algorithm 2). During inference, we process all the notes in the sequence in batches of N_{max} chunks. Each sequence batch, denoted as s_{batch}, is encoded to obtain embeddings $h_{batch} \in \mathbb{R}^{N_{max} \times D}$. Even if the full sequence does not fit into memory, it can be processed in separate batches to obtain all the h_{batch} embeddings. These embeddings are then concatenated along the batch dimension to obtain chunk embeddings for the complete sequence $h \in \mathbb{R}^{N_{total} \times D}$. Finally, the collected embeddings are passed through causal attention and masked multi-head label attention to obtain predictions p based on the complete sequence.

As the computation can be performed in separate batches and then combined, this allows for considerably longer sequences to be used as input during inference. Unlike other methods for extending the context of transformers that rely on reducing or compressing long-distance attention (Beltagy et al., 2020; Munkhdalai et al., 2024), this proposed method is also exact – the result is always the same as it would be with a single pass using infinite memory.

5 Experiment Set-up

5.1 Evaluation framework

We investigate the novel task of temporal ICD code prediction, which requires the prediction of ICD codes at any point during the hospital stay using the notes available at that time, without relying on the discharge summary. To evaluate the performance, we will compare the predictive power of our model at different points throughout the EHR sequence. Our evaluation setup is inspired by the ClinicalBERT model (Huang et al., 2019), which evaluates the likelihood of readmission at different cut-off times since admission.

The cut-off times were selected to be the 25%, 50%, and 75% percentiles of the total volume of notes present in the training dataset, which are shown in Table 2. These correspond to 2, 5, and 13-day cut-offs, respectively. For example, in the *2-day* setting, the model only has access to the notes written in the first 2 days in order to predict all the ICD codes that will be assigned to that patient by the end of their hospital stay. Where space allows, we additionally report on all the notes up to (but excluding) the discharge summary, indicating a setting where the model could be used to assist in the writing of the discharge summary itself. For comparison, we also report performance on the full sequence which includes the discharge summary, although this setting is retrospective and would not provide any predictive benefit. In line with widely used approaches to ICD coding (Mullenbach et al., 2018), we focus on Micro-F1, Micro-AUC and Precision@5 metrics, with additional metrics provided in the appendix.

Percentile	Days elapsed	# notes
25%	1.8	112,594
50%	5.2	225,160
75%	12.8	337,726

Table 2: Percentiles of the total volume of notes present in the training dataset. The number of days corresponding to the 25th, 50th, and 75th percentiles will be used as temporal evaluation points throughout this project.

	# chunks / patient	# patients
2 days	$17.9_{\pm 22.1}$	1,559
5 days	$27.6_{\pm 33.9}$	1,559
13 days	$35.8_{\pm 42.4}$	1,559
excl. DS	$40.4_{\pm 47.7}$	1,559
last day	$48.4_{\pm 48.1}$	1,573

Table 3: Length of EHR in number of chunks per patient (average and standard deviation) and count of patients of our dataset at different temporal cut-offs (dev set).

5.2 Preprocessing

We use the MIMIC-III dataset (Johnson et al., 2016) for evaluation, as it contains a collection of Electronic Health Records with timestamped free-text reports by nurses and doctors, together with the corresponding ICD-9 labels. First, we follow the preprocessing steps outlined by the CAML approach (Mullenbach et al., 2018) to obtain a dataset of free-text clinical notes paired with ICD diagnoses and procedure codes, and we also extract their proposed train/dev/test splits. The label space is vast, so following their method, we focus on predicting the top 50 codes.

For our novel task, we perform some additional preprocessing steps. First, we extract the timestamps of each note and, in cases where the specific time is missing, assign it to 12:00:00 of that day. Moreover, we found that some patients had additional notes beyond the discharge summary document, such as other discharge summaries or nursing notes. We exclude these additional notes to ensure that our EHR sequence always concludes with a single discharge summary document. We also exclude 14 patients as their EHR contains no other notes besides the discharge summary. Table 3 displays the statistics of our dataset at various temporal cut-offs.

5.3 Implementation details

The model is implemented in Pytorch and it was trained on an Nvidia GeForce GTX Titan Xp (12GB RAM) GPU, utilizing an average memory of 11.22 GB. The model processed 5 samples per second and training took an average time of 11 hours and 50 minutes. We used a super-convergence learning rate scheduler (Smith and Topin, 2019), based on its use in HTDS (Ng et al., 2023), and an early-stopping strategy with a 3 epoch patience and a maximum of 20 epochs. Chunk size T was set to 512 tokens as that is the largest size supported by `RoBERTa-base-PM-M3-Voc`. For the main experiments, a limit of $N_{max} = 16$ was used during training, while the entire sequence (with up to 181 chunks) was used for inference. The tuning ranges and chosen hyperparameter values are included in Appendix A.

6 Results

In addition to the LAHST framework described in Sections 3 and 4, we also evaluate PubMedBert-Hier (Ji et al., 2021) and HTDS (Ng et al., 2023) on the early prediction task. HTDS was trained to consider earlier notes in the context while making decisions about the discharge summary, making it the most likely existing model to also perform well on the early prediction task. In addition, HTDS results are very close to the state-of-the-art on the MIMIC-III dataset, making it a very strong baseline. However, HTDS is a larger model and requires considerably more GPU resources compared to LAHST. Therefore, we also report a modified version (HTDS*) which has a comparable number of parameters. We also include the performance of TrLDC (Dai et al., 2022) from the respective paper as an additional strong baseline on classification of the discharge reports.

In Table 4 we report the performance of these systems at increasingly challenging temporal cut-offs. LAHST shows strong performance at any time point, outperforming all the other models at every early prediction task. The results indicate that some of the diagnosis and treatment codes for the whole hospital stay can be predicted already within the first few days of admission. While the performance of all systems is expectedly lower in the more challenging settings, they are still able to reach 46% F1 and 82.9% AUC with only 2 days of information, which could provide useful pre-

	Last day			0-13 days			0-5 days			0-2 days		
Model	**F1**	**AUC**	**P@5**	**F1**	**AUC**	**P@5**	**F1**	**AUC**	**P@5**	**F1**	**AUC**	**P@5**
TrLDC	70.1	93.7	65.9	-	-	-	-	-	-	-	-	-
PMB-H	67.2	91.5	63.0	30.7	68.0	30.2	31.3	68.4	31.0	31.7	68.7	31.5
HTDS	**73.3**	**95.2**	**68.1**	49.7	82.1	47.6	47.5	80.6	45.9	44.5	78.7	43.6
HTDS*	70.7	93.8	66.2	48.6	82.0	47.0	46.7	80.7	45.5	43.6	78.7	43.3
LAHST	70.3	94.6	67.5	**52.9**	**87.0**	**53.0**	**50.3**	**85.4**	**50.7**	**46.0**	**82.9**	**47.1**

Table 4: Evaluation on the early ICD code prediction task at increasingly challenging temporal cut-offs. TrLDC (Dai et al., 2022) result is from the respective paper. We evaluated PubMedBERT-Hier (PMB-H; Ji et al., 2021) and HTDS (Ng et al., 2023) at different early prediction points. HTDS* is a version of HTDS that is more comparable to LAHST in terms of computation requirements. LAHST is the model described in Sections 3 and 4. Results for PMB-H, HTDS, HTDS* and LAHST are averaged over 3 runs with different random seeds.

dictions to the hospital staff. In all the early prediction settings, LAHST achieves the best results according to all metrics. While HTDS is trained to look at earlier documents and is also able to make competitive predictions, it is reliant on information in the discharge summary and therefore underperforms when this is not available. In contrast, the LAHST model is trained to make predictions based on varying amounts of evidence and achieves the best performance.

In the "Last day" setting, which includes the discharge summary, HTDS slightly outperforms LAHST – this is expected, as HTDS is a larger model and specifically trained for discharge summaries. However, when compared to the similarly-sized HTDS*, LAHST delivers comparable F1 along with improved AUC and P@5. Even though LAHST is not trained for this particular setting, the supervision on earlier time points helps it achieve good results also when classifying discharge summaries. In addition, it outperforms both PubMedBERT-Hier and TrLDC according to all metrics. We include larger results tables with additional metrics in Appendix B.

7 Analyzing the Extended Context

Selection of context during inference. The Extended Context Algorithm (ECA) allows the model to include much longer EHR sequences in the context during inference (with a generous 181 chunk cap applied in our experiments). We evaluate the effect of this algorithm compared to alternative strategies used in other hierarchical models. The "Last" setting uses the most recent 16 chunks of text, illustrating the setting where the sequence is truncated from the beginning in order to fit into the model. The "Random" setting samples a random subset of chunks from the sequence instead. The results in Table 5 show that processing the entire sequence with ECA yields substantial performance improvements (+12.7, +13.3 and +15.4 Micro-F1 score for 2 days, 5 days and 13 days) compared to truncating or sampling the sequence. This result highlights the importance of including all the available notes in the input. Only when the discharge note is available (in the *'Last day'* setting) the previous notes become less important and all the strategies give the same performance.

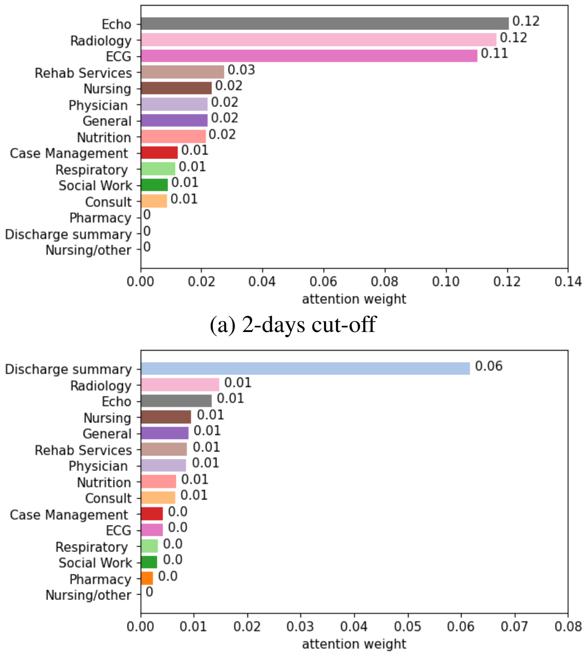
	Last	**Random**	**ECA**
0-2 days	33.5 ±0.6	29.2 ±0.5	**46.2** ±0.1
0-5 days	37.6 ±0.3	35.1 ±0.4	**50.9** ±0.1
0-13 days	38.2 ±0.1	37.7 ±0.5	**53.6** ±0.2
Excl.DS	37.9 ±0.2	38.4 ±0.4	**54.3** ±0.2
Last day	71.0 ±0.3	**71.3**±0.2	71.1 ±0.1

Table 5: Micro-F1 score on the development set, using the LAHST model with alternative strategies for context inclusion.

Selection of context during training. We investigate the effect of randomly sampling different sub-sequences of notes during training. We train an alternative version of LAHST by truncating the sequence to the most recent notes instead of sampling them randomly. During inference, both versions still receive all the notes as input, as described in Algorithm 2. The results in Table 6 show how training without random sampling substantially decreases performance across all evaluation points (-8.4, -8.4, -8.2, -8.0, -0.9, F1 respectively). This indicates that randomly sampling different sub-sequences during optimization augments the training data with

	Last	Random/ECA
0-2 days	37.8 ±0.2	**46.2** ±0.1
0-5 days	42.5 ±0.1	**50.9** ±0.1
0-13 days	45.4 ±0.1	**53.6** ±0.2
Excl. DS	46.3 ±0.1	**54.3** ±0.2
Last day	70.2 ±0.2	**71.1** ±0.1

Table 6: Micro-F1 of LAHST on the development set, using alternative sampling strategies during training.



(a) 2-days cut-off

(b) Last day cut-off

Figure 2: Average attention weight per document type at different temporal cut-offs. The LAHST model processes the complete EHR sequence and focuses more on reports of diagnostic tests for early prediction, switching to the discharge summary when it is available.

different variations which helps the model better generalize to different temporal cut-offs, without increasing memory or computation requirements.

8 Model Interpretability

The attention weights in the label-attention layer of LAHST can potentially be used as an importance indicator of different input notes. A higher weight is associated with an increased relevance of the particular document to predicting a specific code. For an initial visualization, we average the weights across all the codes to find which document types are most important at different temporal cut-offs.

The results are shown in Fig. 2. Within the 2-day cut-off, all the reports that have diagnostic characteristics have received the highest attention weights. For example, the echocardiography report is the description of an ultrasound test to identify abnormalities in the heart structure and is used by cardiologists to diagnose heart diseases (Van et al., 2023). The radiology reports detail the results of imaging procedures such as X-rays and MRIs to diagnose diseases (Alarifi et al., 2021). All of such reports are highly technical and are specifically created to assist physicians with diagnostic practices. In the absence of the discharge summary, they are the most valuable document types for making early predictions of ICD-9 codes and the network has correctly focused more attention on them. In the "Last day" setting, the discharge summary becomes available, containing an overview of the entire hospital stay, and the same model is able to switch most of its attention to it.

9 Conclusions

In this study, we investigated the potential of predicting ICD codes for the whole patient stay at different time points during their stay. Being able to predict likely diagnoses and treatments in advance would have important applications for predictive medicine, by enabling early diagnosis, suggestions for treatments, and optimization of resource allocation. We designed a specialized architecture (LAHST) for this task, which uses a hierarchical structure combined with label attention and causal attention to efficiently make predictions at any possible time points in the EHR sequence. The Extended Context Algorithm was further proposed to allow the model to better handle very long sequences of notes. The system is trained by sampling different sub-sequences of notes, which allows the model to fit into memory while also augmenting the data with variations of available examples. During inference, the whole sequence is then processed separately in batches and combined together with a single attention layer, allowing for lossless representations of very long context to be calculated.

Our experiments showed that useful predictions regarding the final ICD codes for a patient can be made already soon after the hospital admission. The LAHST model substantially outperformed existing approaches on the early prediction task, while also achieving competitive results on the standard task of assigning codes to discharge summaries. The model achieved 82.9% AUC already 2 days after admission, indicating that it is able to rank and suggest relevant ICD codes based on limited information very early into a hospital stay.

10 Limitations

The primary focus of this project was to investigate the feasibility of this novel task and explore a novel architecture for the early prediction of ICD codes. Even though this could open up new avenues for early disease detection and procedure forecasting, our work has some limitations that should be considered in future work.

Firstly, our study is limited to the MIMIC dataset as it is one of the largest and most established available datasets containing electronic health records and ICD codes. However, the findings based on this dataset may not generalise equally to every clinical setting. Therefore, new experiments would need to be conducted on representative data samples before considering applying such technology in practice.

Our experiments focused on PubMedBERT-Hier (Ji et al., 2021), HTDS (Ng et al., 2023) and LAHST. However, there are many other architectures and pre-trained models available which could be investigated in this setting.

Our model is based on a hierarchical transformer architecture which achieves good performance but is also quite computationally expensive compared to LSTM or CNN-based approaches (training the model took roughly 12 hours on a 12GB GPU). With our computational resources, we were limited to running experiments using the [CLS]-token representation and a maximum of 16 chunks in a batch. However, with additional resources this work could be further scaled up by retaining all token representations and increasing the model size to allow for the allocation of additional chunks.

Finally, our evaluation of the temporal ICD coding task is focused on reporting the aggregate metrics for the top 50 ICD-9 coding labels. Future work could investigate a larger number of labels, along with analysing the performance separately on individual labels and label types.

11 Ethics Statement

After careful consideration, we have determined that no ethical conflicts apply to this project. While clinical data is inherently sensitive, it is important to note that the MIMIC-III dataset has undergone a rigorous de-identification process, following the guidelines outlined by the Health Insurance Portability and Accountability Act (HIPAA). This de-identification process ensures that the dataset can be used for research purposes on an international scale (Johnson et al., 2016).

While no conflicts were identified, machine learning systems for ICD coding carry certain risks when deployed in hospitals. Firstly, automated approaches are trained in a supervised manner using data from hospitals, and therefore, they are susceptible to reproducing manual coding errors. These errors may include miscoding due to misunderstandings of abbreviations and synonyms or overbilling due to unbundling errors (Sonabend W et al., 2020). Moreover, automated systems may also suffer from distribution shifts, potentially affecting their portability across various EHR systems in different hospitals (Sonabend W et al., 2020). To address these concerns, it is important to build interpretable models and develop tools that enable human coders to supervise the decisions made by ICD coding models.

Acknowledgements

Mireia Hernandez Caralt acknowledges that the project that gave rise to these results received the support of a fellowship from “la Caixa” Foundation (ID 100010434). The fellowship code is LCF/BQ/EU22/11930076.

References

Mohammad Alarifi, Patrick Timothy, Jabour Abdulrahman, Min Wu, and Jake Luo. 2021. Understanding patient needs and gaps in radiology reports through online discussion forum analysis. *Insights Imaging*, 12(50).

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *CoRR*, abs/2004.05150.

Donna J Cartwright. 2013. Icd-9-cm to icd-10-cm codes: What? why? how? *Advances in Wound Care*, 2(10):588–592.

Hua Cheng, Rana Jafari, April Russell, Russell Klopfer, Edmond Lu, Benjamin Striner, and Matthew Gormley. 2023. MDACE: MIMIC documents annotated with code evidence. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7534–7550, Toronto, Canada. Association for Computational Linguistics.

Krzysztof Choromanski, Han Lin, Haoxian Chen, Tianyi Zhang, Arijit Sehanobish, Valerii Likhosherstov, Jack Parker-Holder, Tamás Sarlós, Adrian

Weller, and Thomas Weingarten. 2021. From block-toeplitz matrices to differential equations on graphs: towards a general theory for scalable masked transformers. In *International Conference on Machine Learning*.

Xiang Dai, Ilias Chalkidis, Sune Darkner, and Desmond Elliott. 2022. Revisiting transformer-based models for long document classification. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7212–7230, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2022. Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM transactions on computing for healthcare*, 3(1):1–23.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019. ClinicalBERT: modeling clinical notes and predicting hospital readmission. *Computing Research Repository*, arXiv:1904.05342v3. Version 3.

Peter B Jensen, Lars J Jensen, and Søren Brunak. 2012. Mining electronic health records: towards better research applications and clinical care. *Nature Reviews Genetics*, 13(6):395–405.

Shaoxiong Ji, Matti Hölttä, and Pekka Marttinen. 2021. Does the magic of bert apply to medical code assignment? a quantitative study. *Computers in Biology and Medicine*, 139:104998.

Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li-Wei H. H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3(160035).

Haanju Yoo Daeseong Kim Sewon Kim. 2022. An Automatic ICD Coding Network Using Partition-Based Label Attention . *SSRN Electronic Journal*.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020. Pretrained language models for biomedical and clinical tasks: understanding and extending the state-of-the-art. *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, page 146.

Fei Li and Hong Yu. 2020. ICD Coding from Clinical Text Using Multi-Filter Residual Convolutional Neural Network. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8180.

Leibo Liu, Oscar Perez-Concha, Anthony Nguyen, Vicki Bennett, and Louisa Jorm. 2022. Hierarchical label-wise attention transformer model for explainable icd coding. *Journal of biomedical informatics*, 133:104161.

Yang Liu, Hua Cheng, Russell Klopfer, Matthew R. Gormley, and Thomas Schaaf. 2021. Effective convolutional attention network for multi-label clinical document classification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5941–5953, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, G.s Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *Proceedings of Workshop at ICLR*, 2013.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. 2018. Explainable prediction of medical codes from clinical text. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1101–1111, New Orleans, Louisiana. Association for Computational Linguistics.

Tsendsuren Munkhdalai, Manaal Faruqui, and Siddharth Gopal. 2024. Leave no context behind: Efficient infinite context transformers with infini-attention. *arXiv preprint arXiv:2404.07143*.

Boon Liang Clarence Ng, Diogo Santos, and Marek Rei. 2023. Modelling temporal document sequences for clinical ICD coding. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1640–1649, Dubrovnik, Croatia. Association for Computational Linguistics.

Leslie N Smith and Nicholay Topin. 2019. Very fast training of neural networks using large learning rate. *Artificial intelligence and machine learning for multi-domain operations applications*, 1106:369–386.

Aaron Sonabend W, Winston Cai, Yuri Ahuja, Ashwin Ananthakrishnan, Zongqi Xia, Sheng Yu, and Chuan Hong. 2020. Automated ICD coding via unsupervised knowledge integration (UNITE). *International journal of medical informatics, 139, 104135*.

Maryam Tayefi, Phuong Ngo, Taridzo Chomutare, Hercules Dalianis, Elisa Salvi, Andrius Budrionis, and Fred Godtliebsen. 2021. Challenges and opportunities beyond structured data in analysis of electronic health records. *Wiley Interdisciplinary Reviews: Computational Statistics*, 13(6):e1549.

Phi Nguyen Van, Hieu Pham Huy, and Long Tran Quoc. 2023. Echocardiography segmentation using neural ode-based diffeomorphic registration field. *IEEE Transactions On Medical Imaging*, XX.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS)*.

Thanh Vu, Dat Nguyen, and Anthony Nguyen. 2020. A label attention model for ICD coding from clinical text. *In Proceedings of IJCAI*. Doi:10.24963/ijcai.2020/461.

Zheng Yuan, Chuanqi Tan, and Songfang Huang. 2022. Code synonyms do matter: Multiple synonyms matching network for automatic ICD coding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 808–814, Dublin, Ireland. Association for Computational Linguistics.

Appendix A

Hyper-parameter	Range
Num. Layers (Mask. Transf.)	**1**,2,3
Num. Heads (Mask. Transf.)	**1**,2,3
Num. Heads (Label Atten.)	**1**,2,3
Peak LR	1e-5, **5e-5**, 1e-4

Table 7: The range of hyperparameters searched for tuning the model. The chosen value is shown in bold.

Appendix B

Detailed results tables using different time cut-offs.

	0-2 days				
	Micro-F1	Macro-F1	Micro-AUC	Macro-AUC	P@5
PubMedBERT-Hier (Ji et al., 2021)	-	-	-	-	-
TrLDC (Dai et al., 2022)	-	-	-	-	-
HTDS (Ng et al., 2023)	44.5	39.7	78.7	77.5	43.6
HTDS*	43.6	40.0	78.7	76.1	43.3
LAHST	**46.0**	**40.1**	**82.9**	**79.5**	**47.1**

	0-5 days				
	Micro-F1	Macro-F1	Micro-AUC	Macro-AUC	P@5
PubMedBERT-Hier (Ji et al., 2021)	-	-	-	-	-
TrLDC (Dai et al., 2022)	-	-	-	-	-
HTDS (Ng et al., 2023)	47.5	42.5	80.6	79.5	45.9
HTDS*	46.7	42.5	80.7	78.2	45.5
LAHST	**50.3**	**44.6**	**85.4**	**82.2**	**50.7**

	0-13 days				
	Micro-F1	Macro-F1	Micro-AUC	Macro-AUC	P@5
PubMedBERT-Hier (Ji et al., 2021)	-	-	-	-	-
TrLDC (Dai et al., 2022)	-	-	-	-	-
HTDS (Ng et al., 2023)	49.7	44.6	82.1	81.2	47.6
HTDS*	48.6	44.6	82.0	79.7	47.0
LAHST	**52.9**	**47.3**	**87.0**	**83.8**	**53.0**

	Excl DS				
	Micro-F1	Macro-F1	Micro-AUC	Macro-AUC	P@5
PubMedBERT-Hier (Ji et al., 2021)	-	-	-	-	-
TrLDC (Dai et al., 2022)	-	-	-	-	-
HTDS (Ng et al., 2023)	50.2	45.2	82.3	81.5	47.8
HTDS*	49.0	45.1	82.4	80.2	47.5
LAHST	**53.5**	**47.8**	**87.3**	**84.1**	**53.7**

	Last Day				
	Micro-F1	Macro-F1	Micro-AUC	Macro-AUC	P@5
PubMedBERT-Hier (Ji et al., 2021)	68.1	63.3	90.8	88.6	64.4
TrLDC (Dai et al., 2022)	70.1	63.8	93.7	91.4	65.9
HTDS (Ng et al., 2023)	**73.3**	**67.7**	**95.2**	**93.6**	**68.1**
HTDS*	70.7	64.9	93.8	91.6	66.2
LAHST	70.3	64.3	94.6	92.6	67.5

Can GPT Redefine Medical Understanding?
Evaluating GPT on Biomedical Machine Reading Comprehension

Shubham Vatsal, Ayush Singh
inQbator AI at eviCore Healthcare
Evernorth Health Services
firstname.lastname@evicore.com

Abstract

Large language models (LLMs) have shown remarkable performance on many tasks in different domains. However, their performance in contextual biomedical machine reading comprehension (MRC) has not been evaluated in depth. In this work, we evaluate GPT on four contextual biomedical MRC benchmarks. We experiment with different conventional prompting techniques as well as introduce our own novel prompting method. To solve some of the retrieval problems inherent to LLMs, we propose a prompting strategy named Implicit Retrieval Augmented Generation (RAG) that alleviates the need for using vector databases to retrieve important chunks in traditional RAG setups. Moreover, we report qualitative assessments on the natural language generation outputs from our approach. The results show that our new prompting technique is able to get the best performance in two out of four datasets and ranks second in rest of them. Experiments show that modern-day LLMs like GPT even in a zero-shot setting can outperform supervised models, leading to new state-of-the-art (SoTA) results on two of the benchmarks.

1 Introduction

Machine Reading Comprehension (MRC) is defined as a task where a system tries to answer a question based on a given context. The context could be anything ranging from a couple of passages to a list of documents. Even though much research has been conducted on MRC, several challenges remain when dealing with MRC tasks (Sugawara et al., 2022), such as the inability to handle long-range dependencies when trying to do reasoning and domain adaptation. Recent improvements in large language modeling has alleviated a lot of the aforementioned issues.

MRC in the biomedical domain (Hermann et al., 2015; Baradaran et al., 2022) has always been a key area of research. Solving a biomedical MRC task faces various challenges including large intricate in-domain vocabulary, dependency on global knowledge, etc. Due to these challenges, there is a wide gap between the performance of conventional methods in the general domain and that of the biomedical domain. Although, traditional machine learning models did show some improvement, they have never been any close to human baselines or gold standards. Contrary to this preconceived notion, modern-day LLMs have shown remarkable performance on many biomedical tasks (Nori et al., 2023; Yang et al., 2023; Cheng et al., 2023).

MRC can have different variations in itself. A contextual MRC requires the LLMs to answer a query solely by relying on a given context. In contrast, a context-free MRC relies on model's embedded knowledge base or any open-source knowledge base, such as Wikipedia, to answer a query instead of using only the context provided. Some of the datasets corresponding to context-free MRC are Zhang et al. (2018); Pal et al. (2022). These datasets are classified under context-free MRC because the given context is not sufficient to answer the questions. There is a definite need to explore the LLM's inherent knowledge base or any other source of knowledge to answer these queries. Similarly, Berant et al. (2014b); Pappas et al. (2018); Zhu et al. (2020) comprise of the datasets for contextual MRC. Again, these datasets have been categorized under contextual MRC because the queries asked can be correctly answered just by looking at the provided context. There is no need to induce any kind of external knowledge in order to answer these questions.

Recent LLMs have attained unprecedented performance in a wide array of natural language processing (NLP) tasks (Chang et al., 2023). Although their performance have been evaluated on a multitude of MRC benchmarks in a context-free setting, their performance in a contextual setting has been

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 256–265
August 16, 2024. ©2024 Association for Computational Linguistics

Dataset	ProcessBank	BioMRC	MASH-QA	CliCR
# QA Pairs	150	6250	3493	7184
Avg Context Length	85	255	863	1461
Max Context Length	266	510	2911	3952

Table 1: Corpus Level Statistics

understudied. In this work, we fill out this missing gap by evaluating GPT (OpenAI, 2023) on standard contextual MRC benchmarks of the biomedical domain. The key contributions are:

1. We evaluate different prompting techniques with GPT on four contextual biomedical MRC benchmarks and report new SoTA results.

2. We propose a novel prompting method *Implicit RAG*. In this method, the LLM is asked to first retrieve the sections or textual extracts from the context that may be relevant to the query and then answer the given query. This technique shows that unlike conventional RAG we no longer need vector databases to store the embeddings of the entire corpus. It further emphasizes that LLMs are capable enough to do the retrieval in one go. Experiments show that this technique is able to achieve the best results in two out of four discussed datasets and ranks second in rest of them.

3. Although machine evaluation is a good measure of performance, it falls short when evaluating artificially generated text (Schluter, 2017), where actual human preferences are significantly superior. Therefore, we report qualitative preference metrics by human experts on the output of our proposed approach *Implicit RAG*. We find that humans agree with the generated outputs most of the time.

2 Related Work

MRC evaluates a system's ability to comprehend and then reason to answer questions over the natural language present in a passage or context. Over the years, quite a few variations of this task have been devised to address and evaluate various aspects of a MRC system namely cloze-style (Hermann et al., 2015; Yagcioglu et al., 2018; Pappas et al., 2018, 2020), multiple-choice (Richardson et al., 2013; Lai et al., 2017; Berant et al., 2014a), extractive (Yang et al., 2015; Trischler et al., 2016; Zhu et al., 2020) and generative QA (Nguyen et al., 2016; Kočiskỳ et al., 2018). In this study, we strive to evaluate three of the above discussed forms of MRC in the biomedical domain which are cloze-style, extractive and multiple-choice using GPT.

In order to elicit an answer from an LLM like GPT, one needs to prompt it in natural language in an optimal manner to retrieve the intended answer. To that end, there has been tremendous development in finding optimal methods for prompting LLMs. The maximum performance boost has been seen from the Chain-of-Thought (CoT) Reasoning (Wei et al., 2022) prompting strategy which asks the LLM to explain how it arrived at the answer. More recently, Analogical Reasoning (AR) (Yasunaga et al., 2023a) has been proposed that achieves drastically better performance than CoT and other prompting techniques. AR works by asking the LLM to reason about a problem by giving analogies which in return forces the model to leverage the global knowledge encoded in it. While prompting methods like CoT and AR improve LLM's performance by exploiting the model's global knowledge embedded in it, there has been an increase in developing novel techniques, especially for cases where the context that needs to be searched through to answer the asked query is huge. The context could be one huge document or a combination of multiple short/long documents. In such scenarios, it is very important to identify only the relevant chunks of the context required for the underlying task and pay attention to them. These emerging methods come under the umbrella of Retrieval Augmented Generation (RAG) (Lewis et al., 2020), which has been shown to improve the performance of LLMs by retrieving contextually relevant information from corpora. The basic methodology behind RAG is to use, embed, and store context in a vector database. These embeddings can then be retrieved based on their semantic similarity to the query.

All the aforementioned prompting methods help interface and contextualize inputs in a better manner for LLMs. While the efficacy of these methods has been seen on several large benchmarks in different domains, however, the degree to which they help in contextual biomedical MRC has been understudied. Mahbub et al. (2022) presented an adversarial learning-based domain adaptation frame-

You are a *{profession}* who is given a *{context_type}* and a corresponding *{query_type}*. Your job is to read the given *{context_type}* and then select the best option from a list of options to answer the *{query_type}*.

The *{query_type}* that needs to be answered is listed below.

{query_type}: *{query_text}*
List of options: *{options}*

Here is the *{context_type}* that needs to be read to select the best option from a given list of options for the *{query_type}*.

### *{context_text}* ###

Figure 1: Basic Prompt Template

work for the biomedical MRC task to address the discrepancies in the marginal distributions between the datasets of the general and biomedical domains. Nori et al. (2023) evaluated GPT on medical competency examinations and benchmark datasets. Even though their work talks about biomedical MRC, it concentrates only on context-free MRC benchmarks. Similarly, Singhal et al. (2023) evaluated Med-PaLM 2 on medical competency examinations and thus focuses on context-free biomedical MRC.

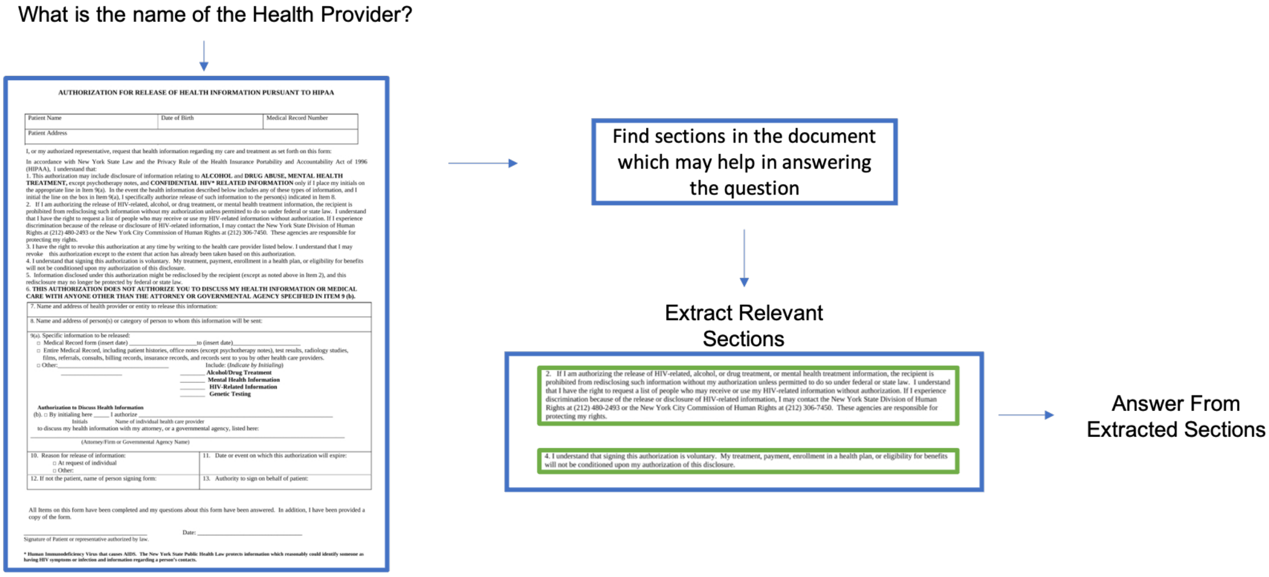
3 Datasets

The four datasets from the biomedical and healthcare domain we choose to explore and analyze the performance of GPT are ProcessBank (Berant et al., 2014b), BioMRC (Pappas et al., 2020), MASH-QA (Zhu et al., 2020) and CliCR (Šuster and Daelemans, 2018). There are a multiple reasons for selecting these four datasets. First, we want to focus on datasets that have not yet been evaluated by modern-day LLMs like GPT. Next, we want to pick up datasets that vary in their statistics and nature. Finally, based on our understanding, these 4 datasets covered the majority of research around contextual MRC in the biomedical field.

ProcessBank contains descriptions of biological processes as context accompanied by multiple-choice questions. BioMRC, an improved version of BioREAD (Pappas et al., 2018) is a large-scale cloze-style dataset. It contains abstracts and titles of biomedical articles and the task of any MRC system is to predict the missing entity in a title using the corresponding abstract as context. In BioMRC, all the biomedical entities mentioned in the abstract are considered as candidate answers and thus one option needs to be chosen from them. MASH-QA is associated with consumer health domain where the answers can consist of sentences from multiple spans of the long context. The candidate answers include every single sentence of the given context. CliCR is again a cloze-style dataset. It contains cloze queries from clinical case reports. Unlike BioMRC, CliCR doesn't really have a list of candidate answers with one of them being the correct answer. Rather, CliCR contains a ground-truth answer set which consists of different lexical and semantic variations of the ground-truth answer, and thus all of them are correct. We use only the test sets of these datasets for all our prompting experiments in a zero-shot setting. We use BioMRC LITE version of BioMRC. The statistics of the four datasets are listed in Table 1.

4 Prompting Techniques

While an exhaustive study of all prior prompting strategies could have been a better experimental setup but due to the cost-prohibitive nature of running large-scale experiments on GPT, we only select the techniques that have shown to perform well in the general domain. Along with these strategies, we also introduce a novel prompting method named *Implicit RAG*. We elaborate on all of these different prompts and their corresponding templates. There may be slight differences in prompt templates of the same prompting strategy across different datasets in order to adhere to the syntactic and semantic rules of English grammar as well as align with the dataset characteristics.



What is the name of the Health Provider?
Find sections in the document which may help in answering the question
Extract Relevant Sections
Answer From Extracted Sections

AUTHORIZATION FOR RELEASE OF HEALTH INFORMATION PURSUANT TO HIPAA

Patient Name	Date of Birth	Medical Record Number
Patient Address		

I, or my authorized representative, request that health information regarding my care and treatment as set forth on this form:

In accordance with New York State Law and the Privacy Rule of the Health Insurance Portability and Accountability Act of 1996 (HIPAA), I understand that:

1. This authorization may include disclosure of information relating to **ALCOHOL** and **DRUG ABUSE, MENTAL HEALTH TREATMENT,** except psychotherapy notes, and **CONFIDENTIAL HIV* RELATED INFORMATION** only if I place my initials on the appropriate line in Item 9(a). In the event the health information described below includes any of these types of information, and I initial the line on the box in Item 9(a), I specifically authorize release of such information to the person(s) indicated in Item 8.
2. If I am authorizing the release of HIV-related, alcohol, or drug treatment, or mental health treatment information, the recipient is prohibited from redisclosing such information without my authorization unless permitted to do so under federal or state law. I understand that I have the right to request a list of people who may receive or use my HIV-related information without authorization. If I experience discrimination because of the release or disclosure of HIV-related information, I may contact the New York State Division of Human Rights at (212) 480-2493 or the New York City Commission of Human Rights at (212) 306-7450. These agencies are responsible for protecting my rights.
3. I have the right to revoke this authorization at any time by writing to the health care provider listed below. I understand that I may revoke this authorization except to the extent that action has already been taken based on this authorization.
4. I understand that signing this authorization is voluntary. My treatment, payment, enrollment in a health plan, or eligibility for benefits will not be conditioned upon my authorization of this disclosure.
5. Information disclosed under this authorization might be redisclosed by the recipient (except as noted above in Item 2), and this redisclosure may no longer be protected by federal or state law.
6. **THIS AUTHORIZATION DOES NOT AUTHORIZE YOU TO DISCUSS MY HEALTH INFORMATION OR MEDICAL CARE WITH ANYONE OTHER THAN THE ATTORNEY OR GOVERNMENTAL AGENCY SPECIFIED IN ITEM 9 (b).**

7. Name and address of health provider or entity to release this information:

8. Name and address of person(s) or category of person to whom this information will be sent:

9(a). Specific information to be released:

□ Medical Record form (insert date) ____________________to (insert date)____________________

□ Entire Medical Record, including patient histories, office notes (except psychotherapy notes), test results, radiology studies, films, referrals, consults, billing records, insurance records, and records sent to you by other health care providers.

□ Other:________________________________ Include: (*Indicate by Initialing*)

____
________ **Alcohol/Drug Treatment**
________ **Mental Health Information**
________ **HIV-Related Information**
________ **Genetic Testing**

Authorization to Discuss Health Information

(b). □ By initialing here _____ I authorize ____
Initials Name of individual health care provider
to discuss my health information with my attorney, or a governmental agency, listed here:
____
(Attorney/Firm or Governmental Agency Name)

10. Reason for release of information: □ At request of individual □ Other:	11. Date or event on which this authorization will expire:
12. If not the patient, name of person signing form:	13. Authority to sign on behalf of patient:

All Items on this form have been completed and my questions about this form have been answered. In addition, I have been provided a copy of the form.

______________________________ Date: ______________________________
Signature of Patient or representative authorized by law.

*** Human Immunodeficiency Virus that causes AIDS. The New York State Public Health Law protects information which reasonably could identify someone as having HIV symptoms or infection and information regarding a person's contacts.**

2. If I am authorizing the release of HIV-related, alcohol, or drug treatment, or mental health treatment information, the recipient is prohibited from redisclosing such information without my authorization unless permitted to do so under federal or state law. I understand that I have the right to request a list of people who may receive or use my HIV-related information without authorization. If I experience discrimination because of the release or disclosure of HIV-related information, I may contact the New York State Division of Human Rights at (212) 480-2493 or the New York City Commission of Human Rights at (212) 306-7450. These agencies are responsible for protecting my rights.

4. I understand that signing this authorization is voluntary. My treatment, payment, enrollment in a health plan, or eligibility for benefits will not be conditioned upon my authorization of this disclosure.

Figure 2: Implicit RAG Technique

Basic The prompt template used for this technique is shown in Figure 1. The Basic prompting approach asks GPT to answer the query in the simplest way possible. The *profession* placeholder specifies the role that GPT has to take in order to answer the asked question. Based on the source of the dataset, this placeholder takes the value of *biologist* in the case of ProcessBank, *biomedical researcher* in the case of BioMRC, *consumer healthcare expert* in case of MASH-QA and *medical expert* in the case of CliCR dataset. Again, based on the source of the dataset, the placeholder *context_type* takes the value of *paragraph* for ProcessBank, *abstract of the paper* for BioMRC, *healthcare article* for MASH-QA and *clinical case report* for CliCR. The placeholder *query_type* takes the value of *query* for ProcessBank, *title containing the missing entity* for BioMRC, *query* for MASH-QA and *query containing the missing entity* for CliCR. The *query_text* placeholder contains the actual text of the query and similarly *context_text* contains the actual text of the context. The *options* placeholder is present only for ProcessBank and BioMRC datasets and contains the choices to select from while answering the asked query.

Chain-of-Thought Reasoning (CoT) The rationale behind using the CoT technique is that there may be multiple smaller questions that need to be answered first in order to conclude the answer of the final asked question. For example, one of the questions asked to GPT is *Has there been at least 6 weeks of provider-directed conservative treatment?*. This question can easily be divided into 3 smaller questions *Has there been any conservative treatment?*, *Was the treatment provider-directed?* and *What was the duration of conservative treatment?*. The prompt template used for this technique is exactly same to that of Figure 1 with an additional line instructing the model to *Think step by step*.

Analogical Reasoning (AR) Inspired by Yasunaga et al. (2023b), we design our own analogical reasoning strategy by tweaking the prompt to fit our problem statement. We do this because, unlike the general domain, GPT would not be able to recall specific dataset-level knowledge, as we are not sure if it was ever trained on the datasets being used in our study. Rather, we frame the prompt so that GPT does not need to rely on a lot on global knowledge. To that end, instead of asking GPT to generate any kind of relevant QA pairs based on its global knowledge, we ask GPT to generate QA pairs from the given context and then answer the initial question. There is one hyperparameter for this technique which is the number of QA pairs to generate.

Implicit Retrieval Augmented Generation (RAG) Most of the work on RAG talks about data retrieval based on an accepted relevancy score and then using LLM prompts to answer the given query. The data retrieval is done by storing the embeddings from some encoder of the entire corpus (a datapoint's context in our case) in a vector database index and then retrieving the most match-

You are a *{profession}* who is given a *{context_type}* and a corresponding *{query_type}*. Your job is to read the given *{context_type}* and then select the best option from a list of options to answer the *{query_type}*.

The *{query_type}* that needs to be answered is listed below.

{query_type}: *{query_text}*
List of options: *options*

Identify *{number_of_sections}* most relevant sections or text extracts from the given *{context_type}* that may help in selecting the best option to answer the given *{query_type}*. The identified sections or text extracts should be distinct from each other. The identified sections or text extracts must be between *{lower_limit_length}* to *{upper_limit_length}* words long.

Now, choose the best option to answer the given *{query_type}* using the identified sections or text extracts.

Here is the *{context_type}* that needs to be read to select the best option from a given list of options for the *{query_type}*.

### *{context_text}* ###

Figure 3: Implicit RAG Prompt Template

ing data points (text extracts or sections from a datapoint's context) for a given query. The key idea behind using RAG is that it helps in saving a lot of computational cost and improves the LLM's performance as now it has to look in a smaller knowledge space to answer the asked question. In our proposed novel prompting technique *Implicit RAG*, we completely ignore the overhead involved in getting the embeddings of the entire corpus and storing them in a vector database. Instead, we ask the LLM itself to find the most relevant text extracts or sections in the given context which may help in answering the asked question, and then later use these extracted sections to conclude the answer to the original question. The general working of our proposed prompting technique is shown in Figure 2. There are two hyper-parameters for this technique. First is, the number of sections to extract, and the next one is the number of words in each section or text extract. The prompt template used for this technique is shown in Figure 3. The hyper-parameter values for the number of sections and number of words in each section is provided in the placeholders *number_of_sections* and *lower_limit_length* and *upper_limit_length*.

5 Results & Analysis

[1]We use the 32k context window version of GPT-4 to conduct all our experiments. We set the temperature, frequency penalty, and presence penalty to 0 and max tokens to 1000 for GPT-4. The results for all the datasets have been discussed individually below. Based on different iterations of experiments, we choose the hyper-parameter number of QA pairs to generate for AR to be 3 for all the datasets. Similarly, for *Implicit RAG*, we choose the hyper-parameter *lower_limit_length* and *upper_limit_length* values as 50 and 200 respectively for all the datasets except MASH-QA. For MASH-QA, we choose *lower_limit_length* as 0 and *upper_limit_length* as 300. We choose *number_of_sections* for *Implicit RAG* to be 1 for MASH-QA, 2 for ProcessBank and 3 for BioMRC and CliCR.

ProcessBank The results for ProcessBank are shown in Table 2. We ran all 4 prompting strategies on the entire test set of 150 datapoints in a zero-shot setup. Every single prompting method outperforms the previously proposed methods, thus giving us

[1] We will release the relevant sections identified by the Implicit RAG technique as well as the question-answer pairs generated by the AR method upon acceptance for all the discussed datasets which can prove useful for other researchers.

Method	Accuracy
Basic (Full)	0.96
CoT (Full)	0.96
AR (Full)	0.96
Implicit RAG (Full)	**0.97**
Implicit RAG (Full)	**0.97**
Gold Structure	**0.77**
ProRead	0.67
SyntProx	0.60
TextProx	0.55
Bow	0.47

Table 2: Results on ProcessBank. The results for Gold Structure, ProRead, SyntProx, TextProx and Bow have been discussed in Berant et al. (2014b)

Method	Accuracy
Basic (1000)	**0.87**
CoT (1000)	0.81
AR (1000)	0.82
Implicit RAG (1000)	0.83
Basic (Full)	**0.87**
MLP-based Weighting	**0.88**
AoA-Reader with BioBERT	0.87
SciBERT-Max-Reader	0.80
AoA-Reader	0.70
AS-Reader	0.62

Table 3: Results on BioMRC. The results for models MLP-based Weighting and AoA-Reader with BioBERT are discussed in Lu et al. (2022) whereas the results for SciBERT-Max-Reader, AoA-Reader and AS-Reader are explained in Pappas et al. (2020)

a new SoTA on this dataset. Among the different prompting strategies, *Implicit RAG* gets the best results. The important observations are:

1. The only 4 to 5 datapoints that GPT got wrong either are very confusing for even humans to answer or had some typo or extra punctuation in ground-truth answers which GPT was not able to mimic during its generation.

2. It is observed that all the GPT prompting strategies work more or less the same if the question can be answered from a small span in the provided context. The reason why *Implicit RAG* is able to outperform other techniques is because this dataset includes around 30% of temporal and true-false type questions which require extensive analysis of the entire context and that the answer can be spread in different segments of the context. Therefore, reducing the knowledge space by extracting relevant sections to answer the asked question helps in improving the performance.

BioMRC The results for BioMRC are listed in Table 3. Due to cost-related reasons, we first compare different prompting methods by running them on a randomly selected 15% (1000 datapoints) subset of the test set and then choosing the best prompting technique to run on the entire test set. All these experiments are done in a zero-shot setting. Amongst the different prompting techniques, Basic prompting gets the best results and *Implicit RAG* ranks second. The important observations are:

1. Even though BioMRC is a cleaner version of BioREAD, there are still elements of lack of structure in the dataset. For example, there is no 1-1 mapping between entity IDs and entities. So, this means the same entity can be mapped to multiple entity IDs and vice versa which causes a lot of confusion when quantifying performance. The authors of BioMRC claim that for any query, the abstract or the context contains all the candidate options including the correct answer but this is not always true leading to more confusion during evaluation.

2. Quite a few times GPT is able to generate an acronym answer instead of its corresponding full form. Ideally, both acronyms and their full forms must be considered as correct answers.

3. There are instances where GPT being a generative model is able to produce semantically similar answers but still they are marked wrong as they do not exactly match with the correct answer. An embedding based metric can be helpful here.

4. There are a lot of entities which are semantically and syntactically the same but still belong to different ontologies and thus have different entity IDs. For example, in one case, GPT generates the answer *amino acids* where the correct answer is *amino acid* but since both of these entities have different IDs, this answer had to be marked wrong.

Another important aspect to note here apart from the lack of structure in the dataset is the overall system design of supervised models which are being compared to GPT when talking about SoTA. Supervised models use 70%-80% of the available data as their training set which allows their parameters to get a good idea about the nuances of the dataset whereas in case of GPT, all our experiments are being conducted in a zero-shot setting. Also since GPT is a generative model, the chances of GPT

Method	EM	F1	P	R
Basic (600)	**0.12**	**0.53**	0.50	**0.56**
Analogical (600)	0.11	0.50	**0.53**	0.47
CoT (600)	0.11	0.52	0.50	0.55
Implicit RAG (600)	0.10	0.52	0.51	0.52
Basic (Full)	**0.14**	**0.53**	**0.50**	**0.57**
Bert	0.9	0.25	0.56	0.16
RoBERTa	0.9	0.29	0.58	0.19
XLNet	0.9	0.29	0.56	0.20
MultiCo	**0.22**	**0.57**	**0.58**	**0.56**
Tanda	0.9	0.25	0.56	0.16

Table 4: Results on MASH-QA dataset. The results for Bert, RoBERTa, XLNet, MultiCo and Tanda have been talked about in Zhu et al. (2020)

Method	EM	F1
Basic (1100)	0.37	0.53
Analogical (1100)	**0.39**	**0.54**
CoT (1100)	0.36	0.51
Implicit RAG (1100)	0.38	**0.54**
Analogical (Full)	**0.34**	**0.52**
Human Novice	0.31	0.45
Human Expert	**0.35**	**0.54**
GA-Anonym	0.25	0.33
GA-Ent	0.22	0.30
GA-NoEnt	0.15	0.34
SA-Anonym	0.20	0.27
Sim-Entity	0.21	0.29

Table 5: Results on CliCR dataset. The results for Human Novice, Human Expert, GA-Anonym, GA-Ent, GA-NoEnt, SA-Anonym and Sim-Entity are explained in Šuster and Daelemans (2018)

generating an answer not present in the candidate answer list despite the final answer being semantically and syntactically the same is really high. But a supervised model is never going to face this problem as it makes its prediction based on the confidence score for each candidate answer and thus the final answer is always going to be present in the candidate answer list.

MASH-QA The results for MASH-QA are shown in Table 4. We start by conducting a comparison between different prompting strategies by evaluating them on a randomly selected 15% (600 datapoints) subset of the test set. These experiments are undertaken in a zero-shot setting. As we can see in Table 4, Basic prompting performs the best while *Implicit RAG* ranks second. The important points to discuss here are:

1. The answers in QA pairs of MASH-QA are very subjective. The authors of this dataset have not specified any structured process that was followed by the healthcare experts when trying to answer the questions asked on a website from where this dataset was sourced in the first place. An in-depth analysis shows that even though GPT is able to extract better answers a lot of times, since it does not match with the ground truth answers, the evaluation metrics do not reflect it's true capabilities.

2. There is a correlation observed between increase in the number of sections and decreasing performance of *Implicit RAG*. The reason is that the answers in this dataset are long span and thus with increasing number of sections, there is a loss of contextual continuity as the ground truth answers can get split across multiple sections. This ends up confusing the LLM making it difficult to choose the right set of sentences from different sections. Hence, it performs the best when asked to extract just one section from the context. But because MASH-QA contains answers which can be present in disjoint spans of the context, extracting just one section is not able to make *Implicit RAG* the best performing prompting method.

3. One question which may arise is whether extracting one section or having the hyper-parameter number of sections set to 1 for *Implicit RAG* makes it same as that of Basic prompting. *Implicit RAG* and Basic prompting become the same only when we are not only extracting just one section from the context but also when the hyper-parameter number of words for *Implicit RAG* is set equal to the length of the entire given context. But for MASH-QA, the hyper-parameter number of words is set to 300 with the lower limit being 0 and upper limit being 300 and hence they are different.

Again, we need to reiterate that GPT's performance in case of MASH-QA is being compared to supervised models which use 70%-80% of the total data as their training set allowing their parameters to capture granular details better than a generic LLM like GPT in a zero-shot setting.

CliCR The results for CliCR can be seen in Table 5. Again, due to cost-related reasons, we first compare different prompting techniques by running them on randomly selected 15% (1100 datapoints) subset of the test set and then choosing the best

Dataset	ProcessBank (50)		BioMRC (50)		MASH-QA (50)		CliCR (50)	
Pattern	✓(46)	✗(4)	✓(41)	✗(9)	✓(7)	✗(43)	✓(31)	✗(19)
Right Section	100%	100%	95%	56%	100%	93%	81%	32%
Wrong Section	0%	0%	5%	44%	0%	7%	19%	68%

Table 6: Qualitative Analysis of *Implicit RAG* on ProcessBank, BioMRC, MASH-QA and CliCR

prompting method to run on the entire test set. All these experiments are done in a zero-shot setting. Amongst the different prompting strategies, *Implicit RAG* and AR get the best results in terms of F1 metric. AR minutely performs better than *Implicit RAG* when compared in terms of the Exact Match (EM) metric. However, EM is a very harsh metric for a generative model as there can be so many possible variations of semantically similar output which are not wrong. Since AR is computationally faster with respect to *Implicit RAG*, we ran AR on the entire dataset. All the prompting methods outperform the previously proposed methods. Not only does GPT surpass the performance of previous models, but it also comes close to Human Expert performance while beating Human Novice results. The important observations are:

1. The authors of CliCR mention that for the training of supervised models, only those instances are used for which at least one ground-truth answer from the set of ground-truth answers occurs in the clinical case report or the context. But for the evaluation part, for both validation and test sets even those datapoints are included where there is no intersection between ground-truth set and the entities mentioned in the context. This favors supervised learning settings as supervised models have a separate training and development set which can allow the parameters of the model to learn such cues. GPT is still able to perform better possibly because the global knowledge embedded in its parameters gives it enough evidence to perform well.

2. The authors of CliCR compare various skills in their work between the previous SoTA (GPT is the new SoTA) model GA-NoEnt and Human Expert and show that there still exists a huge gap between them. Since GPT is able to achieve almost Human Expert level performance, we can expect it to show similar capabilities in other MRC tasks.

3. There are multiple reasons why *Implicit RAG* performs well on this dataset. First, the mean length of context in this dataset is 1461 words which indicates that with increasing size of context, the chances of analysis of different sections of the context simultaneously to answer a question is high and that is the core idea behind *Implicit RAG*. Next, the authors of CliCR list out that 70% of the queries in this dataset require the *bridging* skill, 40% require the skill of *tracking* and around 25% demand the *spatiotemporal* skill. All these three skills indicate that answering queries in this dataset require deriving cues from different segments of the context and that is what we propose as the key rationale behind *Implicit RAG*.

Implicit RAG Out of the four datasets that we use in this study, *Implicit RAG* is able to achieve the best results for two of them when compared with other prompting techniques. It ranks at the second place for the other two datasets. One of the questions which may arise is whether Implicit RAG can be applicable to contexts which cannot fit in LLM's 32k token limit. Implicit RAG will especially perform better than other prompting techniques in cases where the context size is more than 32k. In such cases, we can chunk the context and make multiple calls to Implicit RAG to retrieve relevant sections given the query. Once all the relevant sections have been retrieved, the last call to Implicit RAG can use these sections to arrive to an answer. But all other prompting techniques require analysis of the entire context (greater than 32k in this case) at the same time to arrive to an answer. We further do a qualitative analysis on 50 randomly picked datapoints for all four datasets. We check how many times the extracted sections are relevant to the question or not. Even if 1 out of all the extracted sections are relevant, we consider that to be a valid retrieval irrespective of whether the final answer was correct or incorrect. The results are shown in Table 6. As we can see, *Implicit RAG* is able to extract relevant sections in most cases.

6 Conclusion

In this work, we show that even in a zero-shot setting, GPT surpasses the performance of supervised models for two out of four benchmarks. Furthermore, GPT's performance comes close to that of Human Expert for one of the benchmarks. Our study corroborates that LLMs indeed have surpassed preconceived techniques even on difficult to model domains like biomedicine. We also come up with a novel prompting method *Implicit RAG* which gets the best results in two out of four datasets and ends up at rank two in others. This opens a new research direction for the RAG domain allowing other researchers to experiment with this technique on other domain datasets.

7 Limitations

Due to cost associated with running large-scale experiments with GPT, we did a comparison of different prompting techniques on a subset of about 15% of the entire test set for three out of four datasets we discuss in this work. It could be possible that there may be a slight difference in the distribution of the random subset we chose in comparison to the entire test set and this could potentially change the final results obtained by a given prompting technique although we expect the difference to be small. As discussed earlier, in cases where the answer to a query can be found in a small span of the context, there is not a huge difference between different prompting techniques. Thus, running the Basic prompting method will be computationally more inexpensive than running heavier prompting strategies like AR or *Implicit RAG*.

References

Razieh Baradaran, Razieh Ghiasi, and Hossein Amirkhani. 2022. A survey on machine reading comprehension systems. *Natural Language Engineering*, 28(6):683–732.

Jonathan Berant, Vivek Srikumar, Pei-Chun Chen, Abby Vander Linden, Brittany Harding, Brad Huang, Peter Clark, and Christopher D. Manning. 2014a. Modeling biological processes for reading comprehension. In *Conference on Empirical Methods in Natural Language Processing*.

Jonathan Berant, Vivek Srikumar, Pei-Chun Chen, Abby Vander Linden, Brittany Harding, Brad Huang, Peter Clark, and Christopher D Manning. 2014b. Modeling biological processes for reading comprehension. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1499–1510.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Kaijie Zhu, Hao Chen, Linyi Yang, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2023. A survey on evaluation of large language models. *arXiv preprint arXiv:2307.03109*.

Kunming Cheng, Qiang Guo, Yongbin He, Yanqiu Lu, Ruijie Xie, Cheng Li, and Haiyang Wu. 2023. Artificial intelligence in sports medicine: could gpt-4 make human doctors obsolete? *Annals of Biomedical Engineering*, pages 1–5.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28.

Tomáš Kočiskỳ, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Yuxuan Lu, Jingya Yan, Zhixuan Qi, Zhongzheng Ge, and Yongping Du. 2022. Contextual embedding and model weighting by fusing domain knowledge on biomedical question answering. In *Proceedings of the 13th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pages 1–4.

Maria Mahbub, Sudarshan Srinivasan, Edmon Begoli, and Gregory D Peterson. 2022. Bioadapt-mrc: adversarial learning-based domain adaptation improves biomedical machine reading comprehension task. *Bioinformatics*, 38(18):4369–4379.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human generated machine reading comprehension dataset. *choice*, 2640:660.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023. Capabilities of gpt-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*.

OpenAI. 2023. Gpt-4 technical report.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on Health, Inference, and Learning*, pages 248–260. PMLR.

Dimitris Pappas, Ion Androutsopoulos, and Harris Papageorgiou. 2018. Bioread: A new dataset for biomedical reading comprehension. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Dimitris Pappas, Petros Stavropoulos, Ion Androutsopoulos, and Ryan McDonald. 2020. Biomrc: A dataset for biomedical machine reading comprehension. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, pages 140–149.

Matthew Richardson, Christopher J.C. Burges, and Erin Renshaw. 2013. MCTest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 193–203, Seattle, Washington, USA. Association for Computational Linguistics.

Natalie Schluter. 2017. The limits of automatic summarisation according to ROUGE. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 41–45, Valencia, Spain. Association for Computational Linguistics.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, et al. 2023. Towards expert-level medical question answering with large language models. *arXiv preprint arXiv:2305.09617*.

Saku Sugawara, Nikita Nangia, Alex Warstadt, and Samuel R Bowman. 2022. What makes reading comprehension questions difficult? *arXiv preprint arXiv:2203.06342*.

Simon Šuster and Walter Daelemans. 2018. Clicr: a dataset of clinical case reports for machine reading comprehension. *arXiv preprint arXiv:1803.09720*.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2016. Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models.

Semih Yagcioglu, Aykut Erdem, Erkut Erdem, and Nazli Ikizler-Cinbis. 2018. Recipeqa: A challenge dataset for multimodal comprehension of cooking recipes. *arXiv preprint arXiv:1809.00812*.

Jingye Yang, Cong Liu, Wendy Deng, Da Wu, Chunhua Weng, Yunyun Zhou, and Kai Wang. 2023. Enhancing phenotype recognition in clinical notes using large language models: Phenobcbert and phenogpt. *arXiv preprint arXiv:2308.06294*.

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 2013–2018.

Michihiro Yasunaga, Xinyun Chen, Yujia Li, Panupong Pasupat, Jure Leskovec, Percy Liang, Ed H. Chi, and Denny Zhou. 2023a. Large Language Models as Analogical Reasoners. ArXiv:2310.01714 [cs].

Michihiro Yasunaga, Xinyun Chen, Yujia Li, Panupong Pasupat, Jure Leskovec, Percy Liang, Ed H Chi, and Denny Zhou. 2023b. Large language models as analogical reasoners. *arXiv preprint arXiv:2310.01714*.

Xiao Zhang, Ji Wu, Zhiyang He, Xien Liu, and Ying Su. 2018. Medical exam question answering with large-scale reading comprehension. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Ming Zhu, Aman Ahuja, Da-Cheng Juan, Wei Wei, and Chandan K Reddy. 2020. Question answering with long multiple-span answers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3840–3849.

Get the Best out of 1B LLMs: Insights from Information Extraction on Clinical Documents

Saeed Farzi[*], **Soumitra Ghosh**[*], **Alberto Lavelli** and **Bernardo Magnini**
Fondazione Bruno Kessler, Via Sommarive 18, Povo, Trento, Italy
{sfarzi, sghosh, lavelli, magnini}@fbk.eu

Abstract

While the popularity of large, versatile language models like ChatGPT continues to rise, the landscape shifts when considering open-source models tailored to specific domains. Moreover, many areas, such as clinical documents, suffer from a scarcity of training data, often amounting to only a few hundred instances. Additionally, in certain settings, such as hospitals, cloud-based solutions pose privacy concerns, necessitating the deployment of language models on traditional hardware, such as single GPUs or powerful CPUs. To address these complexities, we conduct extensive experiments on both clinical entity detection and relation extraction in clinical documents using *1B parameter* models. Our study delves into traditional fine-tuning, continuous pre-training in the medical domain, and instruction-tuning methods, providing valuable insights into their effectiveness in a multilingual setting. Our results underscore the importance of domain-specific models and pre-training for clinical natural language processing tasks. Furthermore, data augmentation using cross-lingual information improves performance in most cases, highlighting the potential for multilingual enhancements.

1 Introduction

In the last few years the deep learning revolution has produced significant changes in information extraction (IE) from clinical text. Pre-trained large language models (LLMs) based on attention and transformer architectures (e.g., BERT, T5, etc.) have become popular mainly due to their superior performance with respect to traditional machine learning approaches. Fine-tuning on downstream tasks has been the standard approach used to transfer general pre-trained knowledge to specific tasks of interest, including information extraction from clinical documents. In addition to fine-tuning, continuous pre-training (Gururangan et al., 2020) has shown to be effective to adapt a LLM to the medical domain or to a specific set of languages. Recently, very large language models have further increased both the amount of data used in the pre-training phase, and the complexity of the model parameters. The resulting models (e.g., GPT-4) achieve high performance with few-shot or even zero-shot in-context learning techniques (i.e., prompting) (Liu et al., 2023). Finally, instruction-tuning (Zhang et al., 2023) has emerged as a powerful approach to align pre-trained LLMs to human expectations for a number of natural language processing (NLP) and conversational tasks, further improving usability and performance of LLMs.

Although there is a clear trend towards large language models with general purpose conversational abilities (e.g., ChatGPT), when the choice is constrained to open source models for a domain-specific downstream task (more than often in a low-resource setting), the current landscape of solutions is rather restricted. In addition, there are good reasons to constraint application solutions to small models, particularly because they are computationally manageable, avoiding the need of expensive hardware or to move sensitive data on the cloud. Given the above considerations, there is a lack of consensus on what would be the best solution.

With the aim of shedding light in the current LLM landscape, in this paper we investigate how available small LLMs perform on fine-tuning, continuous pre-training and instruction-tuning on information extraction from clinical documents. We consider LLMs that are available open source, are within the range of 1B parameters, and that are available in several versions, allowing to investigate the impact of multilinguality and instruction-tuning. Our experiments encompass English and Italian datasets for clinical entity detection, and Italian and Spanish for relation extraction, addressing

[*] These authors contributed equally to this work.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 266–276
August 16, 2024. ©2024 Association for Computational Linguistics

two primary research questions:

- Is continuous pre-training, both on languages and domain, effective on our tasks and domains? Does it allow to reduce the need of fine-tuning data in our low-resource setting?
- Is general purpose instruction-tuning effective? Is it competitive with continuous pre-training, both on domain and languages?

In addition to *core experiments* on small LLMs, we conducted additional experiments aiming at assessing the role of data augmentation on the same models and tasks. Data augmentation is a common practice to boost performance, and we are interested in either merging or translating datasets of different languages, as they are becoming more and more available, although in limited amounts.

The primary contributions of the paper include: (i) comparing fine-tuning, continuous pre-training, and instruction-tuning of the same pre-trained model on two NLP tasks, a novel comparison to our knowledge; (ii) investigating the relations between continuous pre-training on languages and continuous pre-training on a specialized domain, suggesting a promising research direction; (iii) demonstrating that, through accurate parameter optimization, language models with 1B parameters remain competitive, although absolute performance was not the primary focus; and (iv) indicating that language-based data augmentation enhances performance in our low-resource setting.

2 Background

2.1 Pre-trained Language Models

In recent years, there has been extensive research on LLMs owing to their capacity for pre-training, allowing them to learn from vast amounts of data in a self-supervised manner. These models have demonstrated remarkable performance across various NLP tasks (Howard and Ruder, 2018; Radford et al., 2019). Scaling LLMs, either by increasing model size or training data, often enhances their capacity for downstream tasks. Several studies have explored the performance limits through scaling, primarily focusing on enlarging model size while maintaining similar architectures and pre-training tasks. Continuous pre-training has emerged as a method to enhance LLM performance in specific domains (Gururangan et al., 2020).

2.1.1 Instruction Tuning

A significant issue with LLMs is the discrepancy between their training objective and users' needs: while LLMs are typically trained to minimize contextual word prediction errors on large datasets, users expect the model to "follow their instructions helpfully and safely". Instruction tuning (Khashabi et al., 2020; McCann et al., 2018) is proposed as a technique to enhance the capabilities and controllability of LLMs. It consists in training LLMs using (INSTRUCTION, OUTPUT) pairs, where INSTRUCTION denotes the human instruction for the model, and OUTPUT the desired output that follows the INSTRUCTION. Instruction tuning bridges the gap between the next-word prediction objective of LLMs and users' objectives of instruction following, thereby increasing controllability and predictability. Additionally, it is computationally efficient and aids LLM adaptation to specific domains.

2.2 LLMs and Information Extraction

Named Entity Recognition (NER) is a key NLP task involving the identification and classification of entities within text. Early methods relied on rule-based systems and manual dictionaries for entity identification (Petasis et al., 2001; Ruokolainen et al., 2020). A significant advancement in NER came with the introduction of Conditional Random Fields (CRFs), which effectively addressed sequence labeling tasks (Lafferty et al., 2001). The emergence of transformer-based models such as BERT (Devlin et al., 2019), GPT-3 (Brown et al., 2020), and their specialized variants has revolutionized NER by capturing contextual information adeptly. These models exhibit remarkable performance across various domains. Furthermore, NER extends to multilingual and cross-lingual scenarios (Zanoli et al., 2023), where models like XLM-R have proven effective (Conneau et al., 2020).

Relation extraction (RE) is concerned with the identification and categorization of relationships between entities mentioned in text (Mintz et al., 2009). In this paper our focus is on the extraction of relationships from clinical documents.

2.3 IE from Clinical Documents

Historically, medical concept extraction began with rule-based systems like MetaMap (Aronson and Lang, 2010) or hybrid systems (rule-based and ML) cTAKES (Savova et al., 2010) but faced challenges with the complexity of clinical text. Dictionary

Lookup, another popular approach (Doğan et al., 2014), relied on exact matching with predefined clinical terms but lacked robustness to term variability. BANNER, an open-source biomedical NER system (Leaman and Gonzalez, 2008), emerged as a domain-independent solution, surpassing baseline systems and serving as a benchmark. Deep learning models, notably CNN-based architectures (Zhu and Wang, 2019), have enhanced NER by leveraging neural networks' sequential insight.

Transformer-based models like BERT (Devlin et al., 2019) have further improved NER performance in clinical documents, adapted by researchers for biomedical NER (Michalopoulos et al., 2021; Lamproudis et al., 2022). XLM-RoBERTa model experimentation on the E3C corpus (Zanoli et al., 2023) showcased its efficacy in various setups. Recent trends show a shift towards employing transformer models in diverse roles, including pipeline systems and Seq2Seq models (Wang and Lu, 2020; Yamada et al., 2020).

The specialized nature of medical texts requires tailored research and model adaptation. Our work addresses this need by focusing on small generative language models, particularly T5 and its variants (Raffel et al., 2020), for NER and RE tasks within clinical documents. These models offer a resource-efficient alternative, aligning with the practical requirements of the medical domain.

3 Core Experiments

In this section, we present core experiments comparing four model versions across two information extraction tasks (NER and RE) in clinical documents with limited training data and across three different languages.

3.1 Experimental Design

We address the relations among fine-tuning, continuous pre-training and instruction-tuning using models with "1B parameters". The experimental design includes: (i) four versions of a "1B parameters" generative model: a base version (T5), a version with continuous pre-training on several languages (mT5), a version with continuous pre-training on the medical domain (MedMT5), and a version which has been instruction-tuned on general NLP tasks (FLAN-T5). Full fine-tuning approaches have been employed to train the language models to tackle NER and RE tasks whereas for Flan-T5, the prompt fine-tuning approach has been adopted. We run the four models on two IE tasks on clinical documents, NER and RE. For each task we provide results both on a dataset with low-resource data and on a dataset with high-resource data. Finally, experiments cover three languages: English, Italian and Spanish.

A core question behind our experiments on small models is the following: does instruction-tuning overcome the need for continuous pre-training (on languages and domain) on our core models applied to our experimental setting?

3.2 Task 1: Clinical Entity Detection

This task consists in identifying relevant clinical entities from clinical texts, such as patient records, medical reports, and clinical notes. Unlike scientific publications, which focus on research findings, clinical notes encompass documents that report various aspects of clinical practice, including the rationale for a clinical visit, descriptions of physical examinations, assessments of the patient's condition, diagnosis, and subsequent treatment plans. For instance, consider a clinical note:

"Patient John Doe, a 45-year-old male, was admitted on July 15, 2023, with complaints of chest pain. He has a medical history of hypertension and diabetes. During the examination, his blood pressure measured 150/90 mm Hg, and his blood glucose level was 180 mg/dL."

Entity detection here targets essential information, including:

- *Patient Information:* "John Doe", "45-year-old male"
- *Admission Date:* "July 15, 2023"
- *Chief Complaint:* "Chest pain"
- *Medical History:* "hypertension", "diabetes"
- *Vital Signs:* "blood pressure 150/90 mm Hg", "blood glucose 180 mg/dL"

We frame the Clinical Entity Detection task as a text-to-text generation task, emphasizing the identification and labeling of textual spans as named entities within a context. We use the following two datasets.

European Clinical Case Corpus (E3C). This is a dataset of clinical cases already published in journals, covering Spanish, Basque, English, French, and Italian (Magnini et al., 2021, 2022). The annotations focus on both clinical entities, specifically disorders, as classified in UMLS taxonomy, and temporal expressions following the THYME standard. For our experiments, we utilize the preprocessed E3C corpus from (Zanoli et al., 2023), conducting experiments on the English (E3C_En) and Italian (E3C_It) datasets, as outlined in Table 1. We acknowledge that the E3C clinical notes are an idealized version of real-world notes, but they offer a privacy-compliant alternative.

NCBI Disease Corpus. NCBI (Doğan et al., 2014) includes 6,892 mentions of disease names and their corresponding identifiers in 793 PubMed abstracts. Categorized mentions allow flexible matching to MeSH and OMIM concepts, while preserving intended meaning. High inter-annotator agreement and low ambiguity make NCBI a strong foundation for machine learning systems, benefiting biomedical knowledge discovery. Table 3 presents the distribution of disease mentions in the NCBI dataset over the training, dev and test sets.

3.3 Task 2: Test-Result Relation Extraction

This task consists of identifying the relations between laboratory tests and their measurements within a clinical note. Building on recent advancements in the field, we approach test-result relation extraction as a form of text summarization, leveraging text-to-text transformer-based models. The key idea is to represent relations as summarized text of a given input as illustrated in Table 2. For the given clinical note, the summarized text is "<EVENT>creatininemia<RESULT> pari o inferiori a 1.5 mg/dl and <EVENT>ipercolesterolemia<RESULT> *280 mg/dl*". We use the following datasets.

CLinkaRT and TESTLINK datasets. We rely on data sourced from the Italian and Spanish sections of the E3C Corpus, respectively, CLinkaRT (Altuna et al., 2023b) and TESTLINK (Altuna et al., 2023a), which introduce relation extraction in the context of clinical cases. Table 2 reports an example extracted from the CLinkaRT dataset (Altuna et al., 2023b), along with its associated relations between medical laboratory tests and their respective results. Each relation comprises an event, the associated result, and their corresponding positions within the text. For example, in the first relation, "creatininemia" is found within positions [286 - 281], with value "pari o inferiori a 1.5 mg/dl." Table 4 reports some statistics about the CLinkaRT and TESTLINK datasets.

	Training		Test	
Language	**Gold**	**Pre-processed**	**Gold**	**Pre-processed**
English	463	437	561	516
French	596	569	731	695
Italian	361	345	508	461*
Spanish	525	509	820	800
Basque	846	835	1064	1054

Table 1: Entity distribution over E3C languages. [*] We found 460 entities in the GitHub link instead of 481 as reported in (Zanoli et al., 2023).

3.4 Models

We focus on "1B parameters" models, because: (i) fine-tuning is manageable with limited computational infrastructure, often available in industry and academy; (ii) inference can be performed without need of dedicated hardware, which is a great advantage when data can not be transferred on the cloud (e.g., hospitals). Although there might be several options (e.g., BERT models), for our experiments we used T5 models (Raffel et al., 2020), because there are several versions available and they show competitive performance. We report the main characteristics of the T5 models in Table 5.

3.4.1 T5

T5 (Text-to-Text Transfer Transformer) (Raffel et al., 2020) employs a transformer architecture with shared encoder-decoder parameters and undergoes pretraining on extensive text data followed by fine-tuning for specific tasks, ensuring versatility across NLP tasks. Unlike BERT (Devlin et al., 2019), which predicts masked words, T5 formulates tasks as text-to-text problems, leading to superior performance across various benchmarks.

3.4.2 mT5

mT5, or "Multilingual Text-to-Text Transfer Transformer" (Xue et al., 2021), is a multilingual variant of the T5 model pretrained in an unsupervised manner on a diverse multilingual corpus, supporting 101 languages. Demonstrating impressive performance on tasks like translation (Patel et al., 2022), lemmatization (Ulčar and Robnik-Šikonja, 2023), and text simplification (Gonzalez-Dios et al., 2022), mT5 showcases its versatility and effectiveness across different language tasks.

3.4.3 MedMT5

MedMT5 (García-Ferrero et al., 2024) is an encoder-decoder model developed by continuing the training of the mT5 (Xue et al., 2021) checkpoints on a medical domain corpus that includes 3B words in four languages (English, Spanish, French,

Example Clinical Note: Il decorso clinico era stato caratterizzato da un rigetto acuto nel primo mese post-trapianto e da alcuni episodi di tachicardia parossistica sopraventricolare negli anni successivi. La funzionalità renale, dopo l'episodio di rigetto, si era stabilizzata su valori di creatininemia pari 0 inferiori a 1.5 mg/dl. L'esame delle urine non aveva mai evidenziato proteinuria. Era presente da anni ipercolesterolemia (280 mg/dl).

	Position of Result	Result	Position of Event	Event
Relation 1	282-310	pari o inferiori a 1.5 mg/dl	286-281	creatininemia
Relation 2	414-422	280 mg/dl	393-411	ipercolesterolemia

Table 2: An example from the CLinkaRT dataset.

Training	Development	Test	Total
5145	787	960	6892

Table 3: Disease mentions distribution in NCBI.

Datasets	Docs	Relations	Unique Events	Unique Results
CLinkaRT-Train (IT)	83	619	344	410
CLinkaRT-Test (IT)	80	612	332	407
TESTLINK-Train (SP)	81	597	317	332
TESTLINK-Test (SP)	80	668	340	421

Table 4: Statistics about the CLinkaRT dataset. IT: Italian, SP: Spanish

and Italian). It is the first open-source text-to-text multilingual model for the medical domain.

3.4.4 FLAN-T5

FLAN-T5 (Chung et al., 2022) is an instruction-tuned language model that excels in NLP tasks by training on diverse instructions, enabling it to handle a wide range of tasks. Mixing zero-shot, few-shot, and chain of thought prompts during training enhances FLAN-T5's performance, even on tasks not seen during fine-tuning, making it excel in both held-in and held-out tasks.

3.5 Experimental Setup

Both for Named Entity Recognition and Relation Extraction the core approach is based on text-to-text generation using the T5 models. The loss functions utilized for both tasks are the standard cross-entropy losses associated with T5 models.

3.5.1 NER Task

We maintained consistent hyperparameters for all models, including a batch size of 4, a maximum token length of 256 for input and output, epochs 30, 0.05 dropout, a warmup ratio of 0.06, and an epsilon of 1e-8 for Adam optimization. The remaining parameters used default values from the SimpleTransformers[1] library, and a seed[2] value of 32 ensured result reproducibility. While hyperparameter tuning was not exhaustive, we explored varying learning rates (1e-4, 2e-4, 3e-4, 2e-5, and, 3e-5). The most suitable learning rate (for all models) was observed to be 1e-4.

3.5.2 RE Task

We adhered to consistent hyper-parameters across all models during training, including a batch size of 2, a maximum token length of 128 for both input and output sequences, a training duration of up to 100 epochs with early stopping, a learning rate of 4e-5, a gradient accumulation step of 4, a dropout rate of 0.1, a warm-up step count of 500, and an epsilon value of 1e-8 for the Adam optimization. Default values from the Hugging Face library were used for the remaining parameters, and a seed value of 42 was employed to ensure result reproducibility.

All models were trained on an NVIDIA A40 GPU with 48 GB GDDR6 memory.

4 Results and Discussion

In this section we present the results of the core experiments on the two tasks, NER and RE.

4.1 Results on NER

Table 6 illustrated the results of our experiments with various T5 models for the NER task on the E3C and NCBI datasets. The performance of T5, mT5, MedMT5, and FLAN-T5 models on various datasets highlights key insights. The base T5 model shows a relatively high recall on E3C-English, indicating it can identify a large number of relevant entities, but its precision is lower, leading to a moderate F1 score. For E3C-Italian, the precision is higher than recall, but the F1 score remains balanced. On the NCBI dataset, T5 achieves strong precision and recall, resulting in a high F1 score. The multilingual mT5 model performs slightly better than T5 in terms of precision on E3C-English, but its recall is lower, leading to a slightly lower

[1] https://simpletransformers.ai/

[2] We chose a single seed to ensure consistent results and simplify model comparisons. We plan to conduct additional experiments with different random seeds.

Model	**Architecture**	**Parameters**	**# languages**	**Data Source**
T5 (Raffel et al., 2020)	encoder-decoder	770M	1 (English)	Colossal Clean Crawled Corpus (C4)
mT5 (Xue et al., 2021)	encoder-decoder	1.2B	101	Multilingual C4 (mC4)
MedMT5 (García-Ferrero et al., 2024)	encoder-decoder	738M	4	Multilingual
FLAN-T5 (Chung et al., 2022)	encoder-decoder	780M	60	473 datasets (SQuAD, MNLI, WMT-16, etc.)

Table 5: Comparison of T5 models.

Models	**E3C**						**NCBI**		
	English			**Italian**			**English**		
	P	**R**	$\mathbf{F}_1$	**P**	**R**	$\mathbf{F}_1$	**P**	**R**	$\mathbf{F}_1$
Baselines									
DLU (Zanoli et al., 2023; Doğan et al., 2014)	37.08	60.08	45.86	48.46	61.52	54.21	21.3	71.8	31.6
CRF (Lafferty et al., 2001)	51.81	30.43	38.34	65.88	42.39	51.59	-	-	-
Inference Method (Doğan et al., 2014)	-	-	-	-	-	-	59.7	73.1	63.7
State-of-the-art									
BANNER (Leaman and Gonzalez, 2008)	-	-	-	-	-	-	83.8	80.00	81.8
XLM-RoBERTa-PL (Zanoli et al., 2023)	45.67	60.66	52.12	60.31	67.39	63.65	-	-	-
XLM-RoBERTa-CL (Zanoli et al., 2023)	40.81	60.27	48.67	43.28	70.00	53.49	-	-	-
T5-Large Family *(ours)*									
T5	51.50	66.47	58.04	63.22	58.91	61.04	85.65	82.88	84.24
mT5	53.68	57.95	55.73	62.53	60.22	61.35	83.72	78.39	80.97
MedMT5	53.94	66.28	59.48	64.74	70.00	**67.29**	86.70	82.99	84.80
FLAN-T5	53.44	69.19	**60.30**	60.79	63.70	62.21	86.80	83.09	**84.91**
mT5 (data augmented)	54.94	51.74	53.29	58.68	61.74	60.17	-	-	-
MedMT5 (data augmented)	55.65	63.95	**59.51**	64.24	71.09	**67.49**	-	-	-

Table 6: Results for Entity Recognition task. DLU: Dictionary look-up. Highest obtained scores among the T5 variants are highlighted in bold.

Models	**CLinkaRT**			**TESTLINK**		
	Italian			**Spanish**		
	P	**R**	$\mathbf{F}_1$	**P**	**R**	$\mathbf{F}_1$
Baselines						
voc. tran. (Altuna et al., 2023b,a)	29.95	31.86	30.88	17.41	30.24	22.10
GPT (Altuna et al., 2023b,a)	29.55	48.73	36.79	25.24	38.29	30.43
mBERT (Altuna et al., 2023b,a)	61.37	64.37	62.83	61.13	60.03	60.57
State-of-the-art						
ExtremITA-T5 (Hromei et al., 2023)	46.82	26.47	33.82	-	-	-
Simple Ideas-BERT (Micluta-Campeanu and Dinu, 2023a)	65.55	60.62	62.99	-	-	-
LinkMed6 (Muñoz-Castro et al., 2023)	-	-	-	46.99	43.26	45.05
Simple Ideas (Micluta-Campeanu and Dinu, 2023b)	-	-	-	71.45	65.57	68.38
T5-Large Family *(ours)*						
T5	53.20	48.03	50.51	58.66	36.97	45.36
mT5	65.72	53.26	58.84	55.96	50.59	53.14
MedMT5	65.22	59.15	**62.03**	62.28	54.64	**58.21**
FLAN-T5	52.99	49.18	51.01	58.03	41.61	48.47
mT5 (data augmented)	69.56	49.67	57.95	66.22	51.94	58.22
MedMT5 (data augmented)	71.72	56.37	**63.12**	70.76	52.54	**60.30**

Table 7: Results on the Relation Extraction task on clinical data, both core and augmented models.

F1 score. For E3C-Italian, mT5 has a more balanced precision and recall, resulting in a similar F1 score to T5. On the NCBI dataset, mT5's performance is slightly lower than T5 in all metrics. The MedMT5 model, pre-trained on medical data, shows improvements over both T5 and mT5. On E3C-English, it achieves higher recall and F1 scores. For E3C-Italian, MedMT5 significantly improves both precision and recall, resulting in the highest F1 score among the models. The performance on the NCBI dataset is also slightly better than T5. The instruction-tuned FLAN-T5 model achieves the highest F1 score on E3C-English, suggesting its effectiveness for this dataset. For E3C-Italian, its performance is slightly lower than MedMT5 but still strong. On the NCBI dataset, FLAN-T5 performs similarly to MedMT5, maintaining high precision and recall.

4.1.1 Discussion

Dataset size is indeed the primary reason behind the observed performance gap between the E3C and NCBI datasets. Additionally, considering our understanding of the two corpora and their annotation strategies, we comprehend that the E3C corpus is much more complex than the NCBI dataset. This complexity arises from the diverse range of medical concepts annotated in the E3C corpus, including disorders such as diseases or syndromes, findings, injuries or poisoning, and signs or symptoms, whereas the NCBI dataset primarily focuses on disease terms.

Fine-tuning, as traditionally done with models like T5 and mT5, continues to be a reliable approach and may yield superior results when abundant data and computing resources are available. The lagging performance of mT5 compared to T5 on English-only datasets can be attributed to T5's specialization and optimization specifically for the English language, which provides it with a distinct advantage.

MedMT5, designed for the medical domain, performs competitively in general NER tasks, suggesting the potential for domain-specific pre-training in specialized areas. Additionally, model versatility, as seen in FLAN-T5 and specialized models like MedMT5, is a key consideration in selecting the most suitable LLM for a particular NLP task.

The observed difference in performance between instruction-tuning and domain-specific pre-training may indeed stem from the size of the inherent pre-training or instruction-tuning datasets used for the models. Specifically, the domain-specific pretrained model (MedMT5) is trained on a larger corpus of Italian data compared to English. In contrast, the instruction-tuned model (FLAN-T5) may have a higher representation of English. This discrepancy in dataset composition could explain the superior performance of the domain-specific pretrained model on the Italian dataset, while the instruction-tuned model excels on the English datasets.

4.1.2 Error Analysis

Our error analysis revealed two key observations: Firstly, models struggle with interpreting abbreviations like "TAO", "NSIAD", "MIC", etc., often mislabeling them as entities or non-entities, indicating challenges in accurately recognizing and interpreting abbreviations. Secondly, the model erroneously labels "metastasis from adenocarcinoma" as a single entity, failing to recognize "metastasis" and "adenocarcinoma" as separate entities, suggesting a lack of contextual understanding and a tendency to group consecutive tokens into a single entity. This tendency to include stop words within entities contributes to a decrement in overall precision. To address these issues, we propose fine-tuning the model on a larger dataset to enhance abbreviation recognition and contextual understanding, along with improving the accuracy of identifying entity boundaries for enhanced precision.

4.2 Results on RE

As reported in Table 7, within the T5 family, MedMT5 models demonstrate a clear superiority over other family members in both languages, with the exception of Italian, where mT5 exhibits a slightly advantage in terms of precision.

4.2.1 Discussion

When comparing MedMT5 with other models, including baselines and state-of-the-art approaches, it is evident that MedMT5 achieves comparable results in terms of F_1 score across both languages. Notably, most models employ data augmentation, including the mBERT-based approach by Altuna et al. (2023b), which utilizes oversampling techniques for relation classification. In contrast, MedMT5 does not employ additional data. ExtremITA-T5, based on IT5 trained on Italian text from the public domain, performs well in certain NLP tasks (Hromei et al., 2023), but falls short compared to MedMT5, especially in the medical domain.

In terms of system complexity, all models follow a dual-model approach, with one model dedicated to named entity recognition and the other to relation classification in a pipeline manner. In contrast, MedMT5 functions as a generative model in an end-to-end manner. Notably, for mention-level relation extraction tasks such as CLinkaRT and TESTLINK, pipeline approaches do not necessitate position determination during post-processing, underscoring a strength of pipeline systems over generative models in these scenarios.

4.2.2 Error Analysis

Analysis of errors in the relation extraction system reveals two primary sources of errors. Firstly, errors stem from relation positioning, where event and result positions are calculated during post-processing. This involves gathering all input sentences and corresponding model-generated responses, determining sentence length, locating event and result positions, and selecting the closest occurrences if multiple exist. Finally, we compute the precise positions of the events and results.

Secondly, errors arise from partially accurate relations, wherein accurate relations contain one or two erroneously generated letters by the model in either the result or event. For instance, in a generated relation: "7,1 mg/dl**e** <– bilirrubina", the letter "e" is generated unnecessarily (the correct relation is "7,1 mg/dl <– bilirrubina"). These types of errors, rooted in data sparsity, significantly impact on system performance. Partially accurate relations are typically encountered in the context of infrequent or rare events or results. As we compare the MedMT5 and mT5 models outputs, it is evident that MedMT5 exhibits a notably lower count of partially accurate relations compared to mT5 models. This implies that the unsupervised learning approach employed by MedMT5 equips the model with certain in-domain lexicons.

Another notable observation in the outputs concerns the impact of input length on performance. Longer input sentences containing numerous relations tend to result in poor performance for most models, with MedMT5 notably outperforming the others. Our experiments involved exploring both longer sentences and sentences split into shorter ones, revealing a significant enhancement in results with shorter sentences. To mitigate this challenge, a potential solution is to implement a sliding window approach on the input to reduce its length. However, the choice of window size becomes a crucial factor, which we plan to investigate in future research efforts.

5 Data Augmentation Experiments

Here we present additional results on two tasks obtained through data augmentation on the core models discussed in section 3. Our aim is to explore potential correlations between the core and augmented models.

5.1 Data Augmentation on NER

To examine how the T5 model's performance is influenced by cross-lingual data augmentation, we trained the mT5 and MedMT5 models on datasets that included both the English and Italian E3C training sets and subsequently evaluated their performance on the English and Italian E3C test sets. We present the results in Table 6.

Data augmentation for mT5 increases precision on E3C-English but reduces recall, resulting in a lower F1 score compared to the non-augmented mT5. For E3C-Italian, the recall improves, and the F1 score remains comparable. Data augmentation enhances precision for MedMT5 on E3C-English and improves recall on E3C-Italian. However, the overall improvement in F1 scores is marginal in both E3C datasets. While data augmentation can improve certain metrics, its impact is mixed and dataset-dependent. It is most beneficial when it enhances recall without significantly compromising precision, as seen with MedMT5 on E3C-Italian.

5.2 Data Augmentation on RE

Using translation data augmentation is indeed a common technique to leverage cross-lingual information and improve the performance of NLP models, including those used in specific domains such as medical. This approach allows models to generalize across languages and learn from diverse multilingual datasets. To expand the training dataset through translation-based data augmentation, the Spanish training data is translated into Italian using Google Translate then it is utilized as augmented data for an Italian task, and vice versa. Table 7 demonstrates a substantial performance boost in terms of precision for both languages. Specifically, there is an enhancement of almost 6 points for Italian and around 8 points for Spanish. This is influenced by two crucial aspects of the datasets. Firstly, the presence of numerous identical relations in both datasets enhances precision. Secondly, the introduction of translation errors in the training data hampers the model's capability to generate rare relations. In summary, the abundance of similar relations contributes to improved precision, while translation errors negatively impact the model's ability to produce less common relations

6 Conclusions

In a world dominated by large language models, this work delves into the efficacy of smaller, domain-specific models in the context of fine-tuning, continuous pre-training, and instruction-tuning on clinical information extraction tasks. Our findings suggest that, while general-purpose instruction-tuning offers versatility, it may not always be as effective as continuous pre-training in domain-specific tasks. We observed instances where instruction-tuning (FLAN-T5) yielded competitive results, but its performance varied across languages and domains. While models like MedMT5 designed for the medical domain outperform general-purpose counterparts in NER and RE, we find that instruction-tuning varies in effectiveness across languages and domains, emphasizing the importance of domain-specific continuous pre-training. This highlights the need for careful consideration when selecting the most suitable approach for a particular NLP task, weighting factors such as data availability, domain specificity, and computational resources. In a landscape where bigger is often seen as better, our work emphasizes the value of smaller, versatile models in scenarios prioritizing data privacy and traditional hardware.

In our future work, we plan to assess parameter-optimized strategies such as PEFT, LORA, QLORA, and LLAMA-Adapter for training larger models on traditional hardware efficiently. This exploration aims to advance model scalability while considering computational constraints, particularly in resource-limited environments.

Limitations of the Study

Concerning relation extraction, our focus was on the CLinkaRT and TESTLINK tasks, which involve identifying test results and measurements and linking them to corresponding textual mentions of clinical laboratory tests. We specifically concentrated on discovering relations between clinical laboratory tests and their results. To the best of our knowledge, there is currently no relation extraction dataset derived from E3C that encompasses a wider variety of relation types.

Regarding the entity detection results, it is important to note that our experimental datasets contain only one type of entity, and thus the reported scores pertain specifically to that entity type. We acknowledge the value of providing results for individual entity types and will consider incorporating this in future iterations of our work.

Furthermore, we agree that variations in T5 models, such as instruction-tuned and domain-specific pre-trained versions, could potentially influence results due to differences in language coverage. While our current evaluation focuses on overall entity detection performance, we acknowledge the potential impact of these variations on the results. In our future work, we plan to conduct a more comprehensive analysis to explore how different T5 models, including instruction-tuned and domain-specific pre-trained variants, perform across various entity types.

Ethical Considerations

The datasets employed in this study, while residing within the clinical domain, do not contain sensitive or personally identifiable information. These datasets are publicly accessible and openly available for research purposes.

Acknowledgments

This work has been partially funded by the European Union under the Horizon Europe eCREAM Project (Grant Agreement No.101057726) and IDEA4RC Project (Grant Agreement No.101057048). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Health and Digital Executive Agency (HADEA). Neither the European Union nor the granting authority can be held responsible for them. This work was supported by the CHISTERA grant of the Call XAI 2019 of the ANR with the grant number Project-ANR-21-CHR4-0002.

References

Begoña Altuna, Rodrigo Agerri, Lidia Salas-Espejo, José Javier Saiz, Alberto Lavelli, Bernardo Magnini, Manuela Speranza, Roberto Zanoli, and Goutham Karunakaran. 2023a. Overview of TESTLINK at IberLEF 2023: Linking results to clinical laboratory tests and measurements. *Procesamiento del Lenguaje Natural*, 71:313–320.

Begoña Altuna, Goutham Karunakaran, Alberto Lavelli, Bernardo Magnini, Manuela Speranza, and Roberto Zanoli. 2023b. CLinkaRT at EVALITA 2023: Overview of the task on linking a lab result to its test event in the clinical domain. In *Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023), CEUR. org, Parma, Italy*.

Alan R Aronson and François-Michel Lang. 2010. An overview of MetaMap: historical perspective and recent advances. *Journal of the American Medical Informatics Association*, 17(3):229–236.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. Ncbi disease corpus: a resource for disease name recognition and concept normalization. *Journal of biomedical informatics*, 47:1–10.

Iker García-Ferrero, Rodrigo Agerri, Aitziber Atutxa Salazar, Elena Cabrio, Iker de la Iglesia, Alberto Lavelli, Bernardo Magnini, Benjamin Molinet, Johana Ramirez-Romero, German Rigau, Jose Maria Villa-Gonzalez, Serena Villata, and Andrea Zaninello. 2024. MedMT5: An open-source multilingual text-to-text LLM for the medical domain. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*.

Itziar Gonzalez-Dios, Iker Gutiérrez-Fandiño, Oscar m. Cumbicus-Pineda, and Aitor Soroa. 2022. IrekiaLFes: a new open benchmark and baseline systems for Spanish automatic text simplification. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339.

Claudiu D Hromei, Danilo Croce, Valerio Basile, and Roberto Basili. 2023. ExtremITA at EVALITA 2023: Multi-task sustainable scaling to large language models at its extreme. In *Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023), CEUR. org, Parma, Italy*.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. UNIFIEDQA: Crossing format boundaries with a single QA system. In *Findings of the Association for Computational Linguistics: EMNLP 2020*.

John D Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289.

Anastasios Lamproudis, Aron Henriksson, and Hercules Dalianis. 2022. Evaluating pretraining strategies for clinical BERT models. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 410–416.

Robert Leaman and Graciela Gonzalez. 2008. BANNER: an executable survey of advances in biomedical named entity recognition. In *Biocomputing 2008*, pages 652–663. World Scientific.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pretrain, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Bernardo Magnini, Begoña Altuna, Alberto Lavelli, Anne-Lyse Minard, Manuela Speranza, and Roberto Zanoli. 2022. European clinical case corpus. In *European Language Grid: A Language Technology Platform for Multilingual Europe*, pages 283–288. Springer.

Bernardo Magnini, Begoña Altuna, Alberto Lavelli, Manuela Speranza, and Roberto Zanoli. 2021. The E3C project: European clinical case corpus. In *Proceedings of the Annual Conference of the Spanish Association for Natural Language Processing: Projects and Demonstrations (SEPLN-PD 2021) co-located with the Conference of the Spanish Society for Natural Language Processing (SEPLN 2021), Málaga, Spain, September, 2021*, volume 2968 of *CEUR Workshop Proceedings*, pages 17–20. CEUR-WS.org.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *Preprint*, arXiv:1806.08730.

George Michalopoulos, Yuanxin Wang, Hussam Kaka, Helen Chen, and Alexander Wong. 2021. Umls-BERT: Clinical domain knowledge augmentation of contextual embeddings using the Unified Medical Language System Metathesaurus. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Marius Micluta-Campeanu and Liviu P Dinu. 2023a. Simple ideas at CLinkaRT: LeaNER and MeaNER relation extraction. In *Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023), CEUR. org, Parma, Italy*.

Marius Micluta-Campeanu and Liviu Petrișor Dinu. 2023b. Simple ideas@ TESTLINK: Relying on finer models. *Procesamiento del Lenguaje Natural*.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011.

Carlos Muñoz-Castro, Andrés Carvallo, Matías Rojas, Claudio Aracena, Rodrigo Guerra, Benjamín Pizarro, and Jocelyn Dunstan. 2023. LinkMed: Entity recognition and relation extraction from clinical notes in Spanish. *Procesamiento del Lenguaje Natural*.

Ajay Patel, Bryan Li, Mohammad Sadegh Rasooli, Noah Constant, Colin Raffel, and Chris Callison-Burch. 2022. Bidirectional language models are also few-shot learners. In *The Eleventh International Conference on Learning Representations*.

Georgios Petasis, Frantz Vichot, Francis Wolinski, Georgios Paliouras, Vangelis Karkaletsis, and Constantine D Spyropoulos. 2001. Using machine learning to maintain rule-based named-entity recognition and classification systems. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, pages 426–433.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Teemu Ruokolainen, Pekka Kauppinen, Miikka Silfverberg, and Krister Lindén. 2020. A Finnish news corpus for named entity recognition. *Language Resources and Evaluation*, 54:247–272.

Guergana K Savova, James J Masanz, Philip V Ogren, Jiaping Zheng, Sunghwan Sohn, Karin C Kipper-Schuler, and Christopher G Chute. 2010. Mayo clinical text analysis and knowledge extraction system (cTAKES): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*, 17(5):507–513.

Matej Ulčar and Marko Robnik-Šikonja. 2023. Sequence-to-sequence pretraining for a less-resourced Slovenian language. *Frontiers in Artificial Intelligence*, 6:932519.

Jue Wang and Wei Lu. 2020. Two are better than one: Joint entity and relation extraction with table-sequence encoders. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1706–1721.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. Luke: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454.

Roberto Zanoli, Alberto Lavelli, Daniel Verdi do Amarante, and Daniele Toti. 2023. Assessment of the e3c corpus for the recognition of disorders in clinical texts. *Natural Language Engineering*, pages 1–19.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. 2023. Instruction tuning for large language models: A survey. *Preprint*, arXiv:2308.10792.

Yuying Zhu and Guoxin Wang. 2019. CAN-NER: Convolutional Attention Network for Chinese Named Entity Recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*.

K-QA: A Real-World Medical Q&A Benchmark

Itay Manes[1], **Naama Ronn**[1], **David Cohen**[1], **Ran Ilan Ber**[1],
Zehavi Horowitz-Kugler[1], **Gabriel Stanovsky**[2]
[1]K Health Inc, New York, NY
[2]School of Computer Science, The Hebrew University of Jerusalem
{itay.manes,david.cohen,ran.ilanber}@khealth.com

Abstract

Ensuring the accuracy of responses provided by large language models (LLMs) is crucial, particularly in clinical settings where incorrect information may directly impact patient health. To address this challenge, we construct K-QA, a dataset containing 1,212 patient questions originating from real-world conversations held on K Health (an AI-driven clinical platform). We employ a panel of in-house physicians to answer and manually decompose a subset of K-QA into self-contained statements. Additionally, we formulate two NLI-based evaluation metrics approximating recall and precision: (1) *comprehensiveness*, measuring the percentage of essential clinical information in the generated answer and (2) *hallucination rate*, measuring the number of statements from the physician-curated response contradicted by the LLM answer. Finally, we use K-QA along with these metrics to evaluate several state-of-the-art models, as well as the effect of in-context learning and medically-oriented augmented retrieval schemes developed by the authors. Our findings indicate that in-context learning improves the comprehensiveness of the models, and augmented retrieval is effective in reducing hallucinations. We will make K-QA available to to the community to spur research into medically accurate NLP applications.[1]

1 Introduction

Recent advancements in large language models (LLMs) have led to a growing interest in their use in the medical domain in patient-facing applications, where LLMs hold the promise of providing laypersons with high-quality advice at a relatively low cost (Singhal et al., 2023; Han et al., 2023). For instance, in response to the question *"What's good for muscular pain?"*, a good patient-facing response may include, in addition to medical information (the name of a muscle relaxant), also the advice *"Seek medical attention if you have numbness or tingling in limbs"*.

However, there is a lack of benchmarks reflecting user needs and corresponding medically-accurate answers to test these models under real-world conditions. Most existing benchmarks assume textbook questions with multiple-choice or span-based answers (Tsatsaronis et al., 2015; Ben Abacha et al., 2017; Jin et al., 2019). In contrast, real-world questions, like *"Is there any way I can get some medicine for cold sore and ulcer that is killing me?"*, often include various interacting medical conditions (*"cold sore"*, *"ulcer"*), use ambiguous, non-medical jargon (*"that is killing me"*), and require long-form, nuanced answers.

In this work, we present K-QA, a medical QA benchmark containing 1,212 deidentified questions asked by real users on K Health,[2] an AI-driven clinical platform with over 8 million unique users. The questions in K-QA were curated from K Health's vast database of patient-physician interactions, aiming to capture stand-alone medical questions. These can be answered solely based on the information provided in the question, and do not require any prior knowledge about the patient's history or demographics. The resulting corpus is diverse and challenging, spanning over 100 different medical conditions (see examples in Figure 1).

To evaluate state-of-the-art models against K-QA, a team of 12 in-house medical doctors invested more than 400 person hours rigorously answering 201 questions from the dataset in a free-text format. Doctors consulted credible medical sources, such as UpToDate[3] and PubMed[4] to provide accurate and scientifically-backed answers. Their answers

[1] The data and the evaluation script are available at https://github.com/Itaymanes/K-QA. Results and model comparisons can be viewed at https://huggingface.co/spaces/Itaykhealth/K-QA

[2] https://khealth.com

[3] https://www.wolterskluwer.com/en/solutions/uptodate

[4] https://pubmed.ncbi.nlm.nih.gov/

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 277–294
August 16, 2024. ©2024 Association for Computational Linguistics

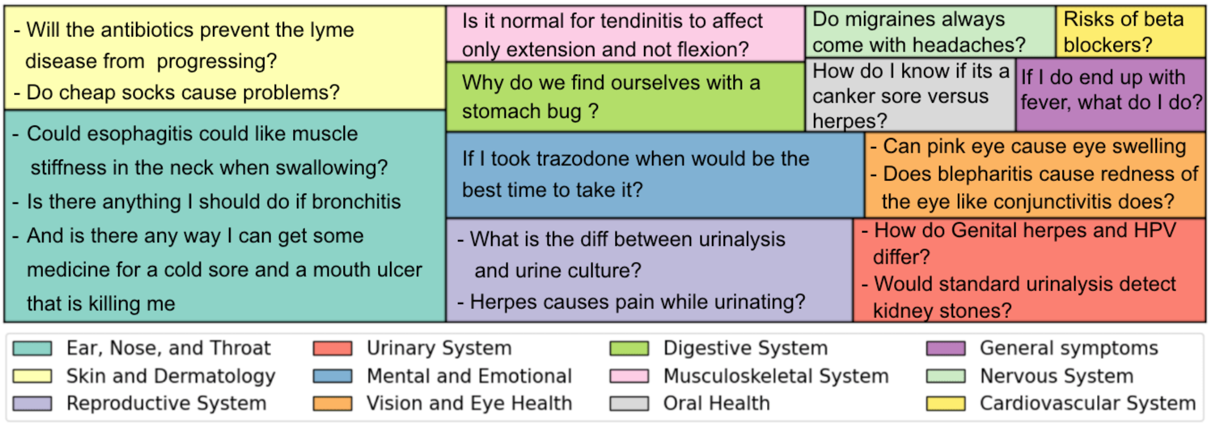


Figure 1: Visualization of K-QA, with box sizes indicating the distribution of patients' reported chief complaints across a wide range of healthcare topics. The questions are open-ended and diverse.

were further reviewed by an experienced overseeing physician. The remaining portion of roughly 1K questions present opportunities for expanding the benchmark through various augmentation techniques, as already demonstrated by Cherian et al. (2024).

To allow fine-grained evaluation, doctors decomposed each answer into an average of roughly 8 minimal semantic content units (Nenkova et al., 2007), resulting in over 1.5K individual statements. In addition, the importance of each statement was manually marked as either (1) *Must Have*, indicating that a model must include this statement in order to be medically accurate (e.g., providing all contraindications for a drug) , or (2) *Nice to Have*, indicating the statement is supplemental in nature (e.g., providing additional conditions where this drug may be helpful).

Following recent work on evaluation of text generation, we use the decomposed ground-truth answers in a natural language inference (NLI)-based evaluation of predicted answers (Honovich et al., 2021; Laban et al., 2022; Aharoni et al., 2023). Concretely, we define two complementing evaluation metrics. First, *comprehensiveness*, which is similar to recall, measures the percentage of ground-truth statements conveyed in the predicted answer. In order to excel in this metric, a model must cover all of the *Must Have* statements annotated by doctors. Second, *hallucination rate*, which is similar to precision, measures how many of all ground-truth statements (either *Must Have* or *Nice to Have*) contradict the predicted answer. To excel in this metric models must not produce *any* medically-inaccurate statements. We find that recent LLMs, like GPT-4, are able to approximate both comprehensiveness and hallucination rate, nearing human assessment of both metrics.

Finally, we evaluate various state-of-the-art LLM-based architectures on K-QA, spanning a wide range of families, including open- and closed-source models, zero-shot vs. in-context learning, and retrieval-augmented generation. We find that all models struggle on *comprehensiveness*, with the best performing model covering only 67.7% of medically-important statements, and while hallucinations seem to decrease with model size and augmented generation, all models still provide medically-dangerous advice in subtle ways which are especially risky for lay users.

We hope that future work adopts K-QA and accompanying metrics as a valuable benchmark to produce medically-accurate NLP applications which can be safely deployed in real-world scenarios.

2 Background

We review existing medical NLP datasets and other recent challenging benchmarks. We highlight key comparisons with K-QA in Table 1.

Medical QA benchmarks. Several diverse health-related question-answering datasets have been compiled, including over biomedical scientific literature (Tsatsaronis et al., 2015; Jin et al., 2019) and medical examinations (Zhang et al., 2018; Pal et al., 2022). The majority of these datasets rely on multiple-choice or span extraction (Jin et al., 2022), which simplify the evaluation process but do not reflect complexity of free-form responses which are often needed in real-world situations (Gehrmann et al., 2023).

Dataset	Consumer Health	Open Domain	Patient-Physician Interaction	Answer Decomposition	Answer Format
BioASQ (Tsatsaronis et al., 2015)	✗	✗	✗	✗	span-based & binary
MedQA (Jin et al., 2021)	✗	✓	✗	✗	multiple choice
LiveMedQA (Ben Abacha et al., 2017)	✓	✗	✗	✓	retrieval
MedicationQA (Abacha et al., 2017)	✓	✗	✗	✗	retrieval
MedQuAD (Ben Abacha et al., 2019)	✓	✗	✗	✗	retrieval
MEDIQA-AnS (Savery et al., 2020)	✓	✗	✗	✗	ranking
HealthSearchQA (Singhal et al., 2023)	✓	✓	✗	✗	-
K-QA (Ours)	✓	✓	✓	✓	long-form generation

Table 1: Comparison between K-QA and previous benchmarks in the field of medical question-answering.

In the context of consumer health questions, our dataset is different from existing benchmarks like MEDIQA-AnS (Savery et al., 2020), LiveMedQA (Ben Abacha et al., 2017) and MedicationQA (Ben Abacha et al., 2019) in several additional key ways. While these datasets source their questions from users searching healthcare websites via the ChiQA system (Demner-Fushman et al., 2020) and retrieve answers through keyword matching, ours originates from authentic patient-physician interactions, ensuring genuine medical inquiries. Furthermore, our dataset includes free-form open-domain responses carefully curated by medical professionals. In addition, the answers in K-QA are segmented into finer atomic statements, enabling fine-grained evaluation.

Challenging LLM benchmarks. Our work joins a recent line of test sets which are challenging for state-of-the-art LLMs, thus enabling further development and experimentation. For example, the Bamboogle benchmark consists of 125 multi-hop questions which stump popular search engines (Press et al., 2022), while the GPQA benchmark contains 445 graduate-level questions in various domains (Rein et al., 2023). K-QA consists of 1,212 questions, as well as a subset of 201 answers, specially curated by in-house physicians.

3 The K-QA Benchmark

In this section, we describe curation and annotation the K-QA dataset, depicted in Figure 2. K-QA consists of two portions - a medium-scale corpus of diverse real-world medical inquiries written by patients on an online platform (Section 3.1) and a subset of rigorous and granular answers, annotated by a team of in-house medical experts (Section 3.2). In Section 3.3, we present an analysis of the dataset, illustrating its medical and linguistic diversity.

3.1 Curating Questions from Real-World Patient-Physician Conversations

All of the questions in K-QA originate from de-identified real-world text-based conversations in English held on a proprietary online medical platform. These conversations contain a wide variety of user intents, such as billing inquiries or prescription renewals, alongside a wealth of queries on varied medical subjects (see Figure 1).

Our goal in creating K-QA is to extract from this large and noisy corpus a diverse dataset of *medical* questions which can be used to test automated models' ability to provide factual and comprehensive medical answers. In particular, we aim for the extracted questions to be as *stand-alone* as possible, without relying on the patient's medical record or the context of the medical discourse. For example, K-QA includes questions such as *"How do Genital herpes and HPV differ?"* (adapted from Figure 1), while we omit questions such as *"Can this allergic reaction be related to my age?"* which assumes prior knowledge about the patient and their previous symptoms.

To achieve this, we performed a rigorous manual annotation, aided by a preliminary automatic preprocessing step. First, we used an open-source BERT-based classifier (Devlin et al., 2018), fine-tuned for distinguishing questions from statements, such as *"sounds like hives to me"*.[5] Next, we ap-

[5] https://huggingface.co/mrsinghania/

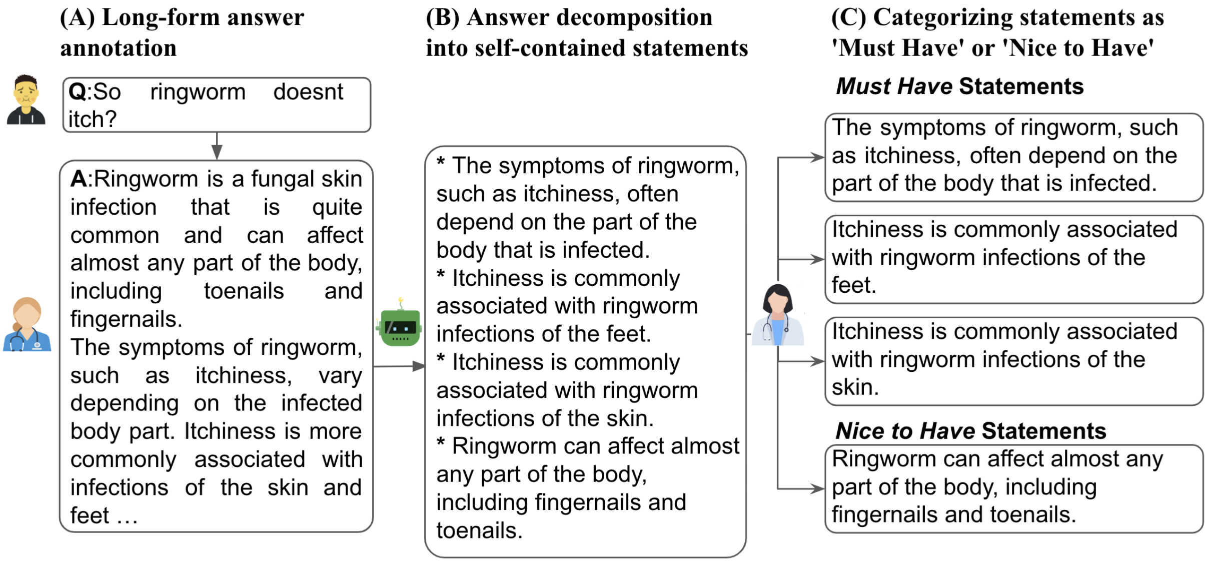

(A) Long-form answer annotation
(B) Answer decomposition into self-contained statements
(C) Categorizing statements as 'Must Have' or 'Nice to Have'
Q:So ringworm doesnt itch?
A:Ringworm is a fungal skin infection that is quite common and can affect almost any part of the body, including toenails and fingernails.
The symptoms of ringworm, such as itchiness, vary depending on the infected body part. Itchiness is more commonly associated with infections of the skin and feet …
* The symptoms of ringworm, such as itchiness, often depend on the part of the body that is infected.
* Itchiness is commonly associated with ringworm infections of the feet.
* Itchiness is commonly associated with ringworm infections of the skin.
* Ringworm can affect almost any part of the body, including fingernails and toenails.
Must Have Statements
The symptoms of ringworm, such as itchiness, often depend on the part of the body that is infected.
Itchiness is commonly associated with ringworm infections of the feet.
Itchiness is commonly associated with ringworm infections of the skin.
Nice to Have Statements
Ringworm can affect almost any part of the body, including fingernails and toenails.

Figure 2: High-level description of the annotation of the K-QA dataset, starting with a physician's considered offline response to an actual patient query obtained from patient-physician interactions. We then use LLMs to decompose the response into self-contained statements, subsequently reviewed and categorized by a panel of medical experts as *Must Have* or *Nice to Have*. The example was simplified for presentation purposes.

plied regular expressions to filter questions about logistics (e.g., billing or delivery instructions). This preprocessing yielded roughly 26K questions, each individually assessed by a medical professional to identify those suitable as stand-alone questions. The dataset comprises diverse questions, each paired with the medical condition diagnosed by the physician at the end of their interaction with the patient, according to ICD-10 conventions (WHO, 1993). For example, the question in Figure 1 is classified as *"Dermatitis, unspecified"*.

3.2 Annotating Granular Physician Answers

We provide comprehensive and granular answers for a diverse subset of K-QA questions, annotated in three steps by a team of 12 in-house medical doctors. This subset enables us to automatically compare different LLMs against high-quality expert answers.

Step 1: Long-form answer annotation. In the first annotation step, exemplified in Figure 2(A), six medical physicians were tasked with providing free-form responses to different sets of questions from K-QA, while an additional physician reviewed their answers and advised where needed. Overall, the first step required roughly 400 skilled person hours (at a cost of roughly 26K USD, based on average physician hourly pay in the U.S.), during which 201 questions from K-QA were answered. Each physician was granted unlimited time and access to reputable medical resources such as UpToDate and PubMed for referencing purposes. Figure 3 depicts the distribution of used sources. Notably, they were explicitly instructed to *avoid* using any generative language models or services. To best emulate the requirements from a user-facing model in the medical domain, annotators were further instructed to write answers tailored for a lay audience seeking consumer-health information. For example, note how the answer in Figure 2 regarding ringworm strays from medical jargon.

Step 2: Answer decomposition into self-contained statements. Following literature on the evaluation of text generation via minimal semantic content units (Nenkova et al., 2007; Liu, 2022), we guided annotators to decompose answers into self-contained statements. Each statement is expected to capture a distinct fact and include sufficient context for independent evaluation. Answer decomposition is presented in Figure 2(B), illustrating the decomposition of a natural answer into atomic statements.

This step was carried out by a panel of 6 medical doctors (distinct from the annotators in the first step) who deconstructed each answer into individual statements. To assist in this process, the panel utilized GPT-4 with a few-shot prompt suggesting potential answer decompositions. The full prompt is provided in Appendix D.1. The annotators could

asr-question-detection

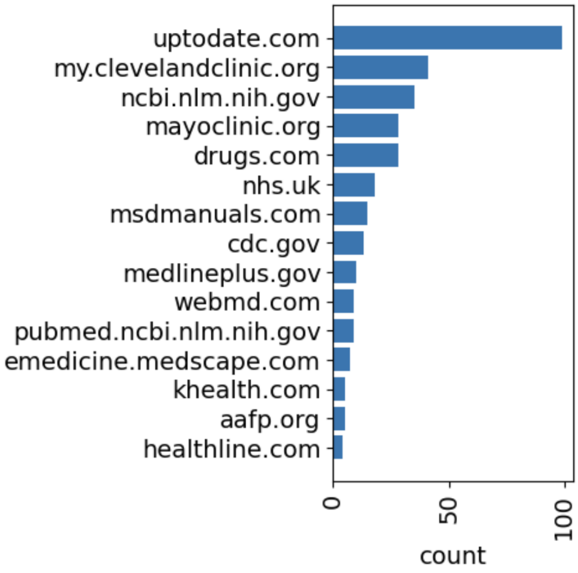


Figure 3: The 15 most used medical resources by the annotators during the curation of the long-form answers.

amend or remove noisy statements, as well as add any missing statements, which they did for 6.86% of the automatically generated statements. In total, this process yielded 1,589 annotated statements, averaging roughly 7.9 statements per answer. The completion of this phase required a total of approximately 30 hours at the cost of approximately 2K USD.

Step 3: Categorizing statements as *Must Have* or *Nice to Have*. At the last step, we asked the same group of medical professionals from step 2 to classify each statement into one of two categories: (1) *Must Have* – facets of the answer which are crucial to convey to a patient when providing medical advice; or (2) *Nice to Have* – statements which are supplemental or informative, but not clinically crucial. For example, as can be seen in Figure 2(C), the long-form answer regarding the itchiness of ringworm was decomposed into four statements, three of which were deemed as *Must Have*, while a statement which provided information about ringworm which is not related to its itchiness was deemed as *Nice to Have*.

A more complex example of medically oriented statement decomposition is presented below. In response to a question about treating hypertension in diabetic patients, the physician recommends ACE inhibitors or ARBs, either alone or in combination with other drugs like calcium channel blockers and thiazides. This scenario illustrates an exclusive OR relation often observed in medical contexts, where multiple treatments are optional but not advised together. To address the dependency between treatment options, we use our statement classification into *Must Have* and *Nice to Have* to preserve the physician's intention, emphasizing the importance of taking either ACE or ARBs and suggesting an additional optional treatment for each. This results in one *Must Have* statement: *"A recommended treatment includes either ACE or ARBs, but not both."*, and two *Nice to Have* statements: (1) *"ARBs can be taken alone or with other medications, such as calcium channel blockers and thiazides."*; and (2) *"ACE can be taken alone or with other medications, such as calcium channel blockers and thiazides."* The full annotation guidelines and more examples are provided in Appendix B.1.

	Count	# Words (avg.)
Questions	1,212	10.06
Answers	201	88.52
Statements		
Must Have	892	14.9
Nice to Have	697	13.74

Table 2: Statistics of the K-QA benchmark.

This process was carried out collaboratively, facilitating discussions within the medical group to collectively assess and reach consensus regarding the perceived levels of importance, amounting to approximately 20 person hours, at the cost of roughly 1.5K USD.

The categorization into *Must Have* and *Nice to Have* represents a discrete approach to assigning importance to statements. However, in a broader context, this method can be extended to assign various weighted scores to each statement and each metric.

3.3 Dataset Statistics

K-QA is derived from a diverse group of 1,055 unique users featuring 1,212 questions, including 201 answers meticulously curated by physicians. Table 2 shows detailed statistics on statements and word counts, while information on the distribution of age and biological sex among users can be found in Table 3.

The questions in our dataset address a wide array of health concerns, as evidenced in Figure 1, covering 172 different medical conditions, according to the ICD-10 system. The diversity in questions is highlighted in Figure 4, showing the top five

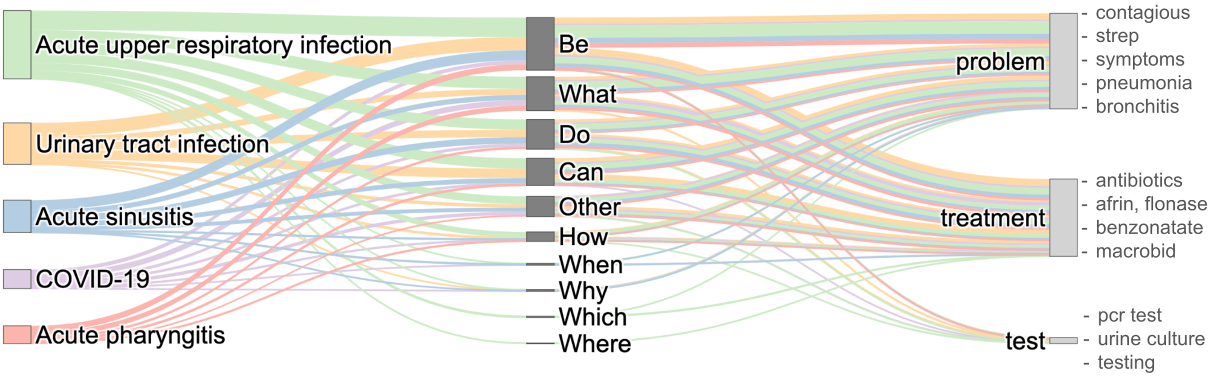


Figure 4: Distribution of the top 5 most prevalent medical conditions, the types of questions related to each condition, and the frequencies of clinical entities within those questions. On the far right, the text most frequently matched with the clinical entities is displayed.

Age Group	Sex (% of users)	
	Female	Male
18-25	9.09	6.67
26-45	36.97	30.30
46-60	9.70	3.64
60+	2.42	1.21

Table 3: Distribution of the users in K-QA by age group and biological sex.

most prevalent medical conditions and their distributions across various question types, such as Be, WH-questions and other forms, as well as various clinical categories (*problem*, *treatment*, and *test*) classified by an open-source, fine-tuned named entity recognition BERT-based model.[6] For example, a question like *"Is it usually normal for someone to have side effects when first starting vitamins?"* is labeled with both *problem* ("side effects") and *treatment* ("vitamins").

4 Evaluation Metrics for K-QA

To evaluate models against K-QA, we propose a natural language inference (NLI; Dagan et al., 2005; Bowman et al., 2015) framework, following recent text generation evaluation (Honovich et al., 2021; Laban et al., 2022; Aharoni et al., 2023).

Our evaluation framework is inspired by FActScore (Min et al., 2023), a metric that measures factual *precision* by computing the percentage of *atomic facts* in a generated answer supported by a reliable external source. Unlike FActScore, which automatically generates statements and assigns them equal importance, our approach involves predefined medical statements with varying clinical significance. This modification enables us to extend the framework and establish a proxy metric for factual *recall* as well.

We consider a predicted answer as a *premise* and each ground-truth statement derived from an annotated answer as an *hypothesis*. Intuitively, a correctly predicted answer should entail every ground-truth statement. This formulation aims to quantify the extent to which the model's answer captures the meaning of the gold answer, abstracting over the wording chosen by a particular expert annotator.

As formulated below, we devise two NLI-based metrics: *comprehensiveness* and *hallucination rate*. These adapt the evaluation of text generation to the medical domain by taking into account K-QA's annotation of *Must Have*, i.e., clinically crucial facets of information, and *Nice to Have* statements, which are supplemental in nature. Both metrics were aggregated across all assessed questions, where higher values of the comprehensiveness metric and lower values of hallucination rates indicate better performance. Figure 5 provides an example illustrating the complete process of evaluating a generated answer and deriving these metrics.

Formally, let $\hat{P}$ denote the model's predicted answer, *Must Have* represents the set of ground-truth statements marked as crucial, *Nice to Have* represents the set of ground-truth statements marked as supplemental, and $S = \textit{Must Have} \cup \textit{Nice to Have}$ is the set of all statements in the gold reference answer.

Comprehensiveness metric. This metric measures how many of the clinically crucial claims are

[6] https://huggingface.co/samrawal/bert-base-uncased_clinical-ner

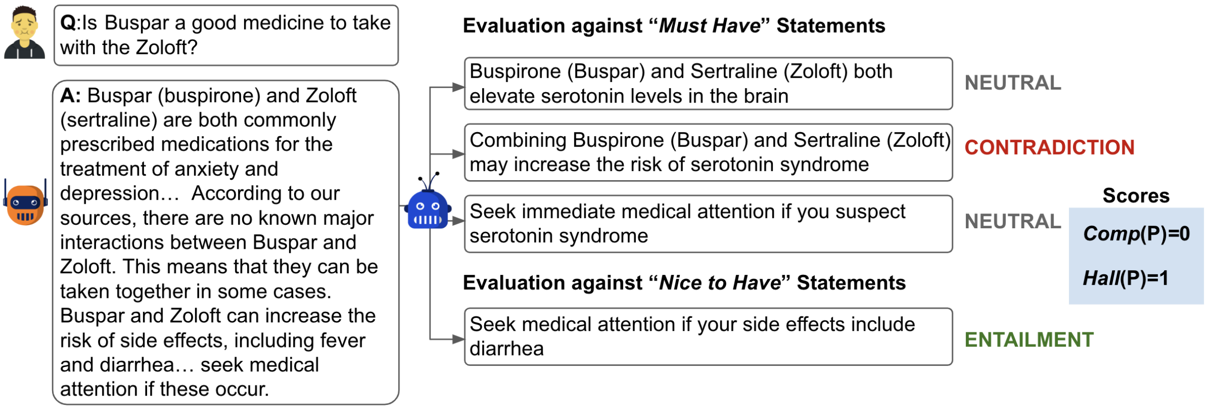

Q:Is Buspar a good medicine to take with the Zoloft?
A: Buspar (buspirone) and Zoloft (sertraline) are both commonly prescribed medications for the treatment of anxiety and depression… According to our sources, there are no known major interactions between Buspar and Zoloft. This means that they can be taken together in some cases. Buspar and Zoloft can increase the risk of side effects, including fever and diarrhea… seek medical attention if these occur.
Evaluation against "Must Have" Statements
Buspirone (Buspar) and Sertraline (Zoloft) both elevate serotonin levels in the brain
NEUTRAL
Combining Buspirone (Buspar) and Sertraline (Zoloft) may increase the risk of serotonin syndrome
CONTRADICTION
Seek immediate medical attention if you suspect serotonin syndrome
NEUTRAL
Evaluation against "Nice to Have" Statements
Seek medical attention if your side effects include diarrhea
ENTAILMENT
Scores
Comp(P)=0
Hall(P)=1

Figure 5: An example of the evaluation procedure, starting with a patient's question and a generated answer from a language model. Each statement is then automatically tested in an NLI framework to determine its relationship to the generated answer. Finally, our metrics are computed, where $Hall(\hat{P})$ counts the number of contraditions (1), and $Comp(\hat{P})$ is equal to 0, because none of the *Must Have* statements were entailed. Different robot symbols signify different models, and the example was simplified for presentation.

included in the predicted answer.

$$Comp(\hat{P}) = \frac{|\{x \in Must\_Have | \hat{P} \text{ entails } x\}|}{|Must\_Have|} \tag{1}$$

I.e., similarly to recall, $0 \leq Comp(\hat{P}) \leq 1$ measures how many ground-truth statements were conveyed in the predicted answer. We particularly focus on those statements marked as crucial by medical experts and do not penalize models for not covering supplemental statements, as these may be somewhat open-ended and arbitrary.

Hallucination rate. This metric measures how many of the ground-truth statements contradict the model's answer.

$$Hall(\hat{P}) = |\{x \in S | \hat{P} \text{ contradicts } x\}| \tag{2}$$

I.e., $Hall(\hat{P}) \in \{0, 1, ..., |S|\}$ penalizes answers that contradict any of the ground-truth statements and hence discourages models from making any sort of false medical statements. Similar to precision, a model can trivially achieve a perfect hallucination score by generating an empty answer $\hat{P} = \emptyset$ since, by definition, no hypothesis contradicts an empty premise.

Automatic evaluation. Following work on NLI-based evaluation, we approximate the metrics above via an automated NLI model. We used GPT-4 in conjunction with few-shot Chain-of-Thought (CoT; Wei et al., 2022) prompt that generates sequential intermediary text representations. The full prompt can be found in Appendix D.2. To assess the quality of the evaluation framework, we randomly selected 50 pairs of questions and their corresponding generated answers from the models described in Section 5.1. This process yielded 398 unique statements pertaining to the specified set of 50 questions.

Three physicians received instructions on how to classify the logical relationship for each triplet (*question*, *answer*, *statement*) into one of three NLI categories. The inter-agreement among annotators was assessed using Fleiss' kappa (κ; Fleiss, 1971) and pairwise agreement. For the three human annotators, the pairwise agreement was 83.2%, and the κ was calculated to be 0.70, signifying moderate to substantial agreement among raters. The agreement with the majority vote of the annotators and the automated model was 83.0%, indicating that the model can perform at a level comparable to human annotators for this complex task.

5 Evaluating State-of-the Art Models

Following the creation of K-QA and the formulation of evaluation metrics, we turn to evaluate the current state of the art in this challenging task.

5.1 Experimental Setup

Models. We use K-QA to evaluate the medical capabilities of 7 recent LLM-based models from diverse families and model sizes. Specifically, we evaluate two 7B instruction-tuned open access models: Mistral (Jiang et al., 2023), and MedAlpaca (Han et al., 2023) which was built upon LLaMA (Touvron et al., 2023) and trained specifically for biomedical tasks, three recent closed instruction-tuned LLMs: Open AI's GPT-3.5 and GPT-4 (Brown et al., 2020; OpenAI, 2023), and Google's PALM-2 (Anil et al., 2023), and finally two recent commercial closed generation search engines: BARD,[7] and Bing Chat.[8] We use zero temperature sampling for all models, except for BARD and Bing Chat, which do not allow setting temperature.

Retrieval augmented generation (RAG). We note that BARD and Bing Chat differ from the other 5 models in our evaluation in that they can reportedly augment their prompt with content retrieved from external sources, albeit in an undisclosed manner. To examine the effect that retrieved content may have on the performance of the other models, we implement Retrieval Augmented Generation approach (RAG; Lewis et al., 2020), which produces responses by conditioning the language model on both the input query and retrieved content. To achieve this, we index publicly available medical documents aimed at the lay audience (such as MayoClinic[9] and NHS[10]) aiming for medical-specific RAG. All the documents in this RAG are publicly available, which is distinct from the primary sources that the physician annotators used to create their answers (Figure 3).

Prompts. Most of our evaluations use the same vanilla zero-shot prompt without prompt engineering which only presents the question, without any additional instructions. In addition, for some models we also report results on another empirically engineered prompt which includes three in-context examples, to explore some of the effect that in context learning (ICL) may have on performance. We did not extensively explore ICL across all models due to various constraints, including MedAlpaca's limited context window, and determining optimal prompts for specific tasks and models remains an open research question (Sclar et al., 2023; Mizrahi et al., 2024). Consequently, we focused on the models that demonstrated the best performance, which are GPT-3.5 and GPT-4. Further exploration of prompt optimization in our case presents an interesting avenue for future work.

Model	*Comp* ↑	*Hall* ↓	*%resp*
MedAlpaca 7B	31.4	56.7	100
Mistral 7B	47.6	28.4	100
PALM-2	50.8	31.3	100
BARD†	62.5	28.4	95.0
Bing Chat†	57.3	25.9	99.5
GPT-3.5	56.2	27.9	100
GPT-3.5+ICL	59.5	23.4	99.5
GPT-3.5+RAG†	50.5	17.9	89.0
GPT-3.5+ICL+RAG†	62.9	**15.4**	96.0
GPT-4	57.5	23.9	100
GPT-4+ICL	**67.7**	25.4	100
GPT-4+RAG†	52.2	22.9	91.5
GPT-4+ICL+RAG†	65.2	24.4	100

Table 4: Model performance on K-QA according to the comprehensiveness and hallucination rate metrics. ICL represents the addition of three in-context examples, and RAG is a medical retrieval augmented setup, as detailed in Section 5.1. The performance of the highest scoring model appears in **bold** for each metric. *%resp* indicates the percentage of questions answered by each model. †Marks models which have a retrieval component.

5.2 Results

The results for all models are shown in Table 4, in terms of the comprehensiveness and hallucination rate metrics defined in Section 4. Below, we highlight key observations based on these results.

Attaining high comprehensiveness is challenging even for state-of-the-art models. Across all models and prompts, the comprehensiveness metric (*Comp*) consistently remains below 68%. This is evident even in cases where models generated longer texts, as seen in the BARD model, which emitted nearly three times as many words per answer (242.1) compared to the physician's response (88.36 words). This underscores the models' difficulty in capturing what physicians consider critically important. Additionally, ICL improves comprehensiveness by instructing the model to include elements beyond a direct response to the question, such as assuming underlying medical concerns in patient inquiries.

While hallucinations seem rare, they could potentially lead to unintended and unsafe medical recommendations. The minimal hallucination rate, achieved by GPT-3.5+ICL+RAG, represents a contradiction of roughly 30 statements out of 1500 annotated examples. Some of the hallucinations may lead to subtle yet dangerous advice. For example, in Figure 5, the physician's statement asserts

[7] https://bard.google.com/

[8] https://www.bing.com

[9] https://www.mayoclinic.org/

[10] https://www.nhs.uk/

that *"Combining Buspar and Zoloft may increase the risk of serotonin syndrome"*, in contrast, the model claims that *"there are no known major interactions between Buspar and Zoloft"*. Finding cause of error in such cases is hard, and physicians are also prone to making dangerous errors. This particular error can be attributed to a combination of missing information within publicly available medical sources and by the LLM assuming that their omission implies the drug combination is safe.

For the GPT models in our evaluation, it seems that larger models lead to improved comprehensiveness, yet the larger the GPT model, the more it seems to introduce new hallucinations. In Table 4, we observe that GPT-4 outperforms GPT-3.5 under every comparable setting. However, the improved comprehensiveness comes at the cost of an increase in the hallucination rate.

Domain-specific RAG reduces hallucinations. Among all configurations, GPT-3.5+ICL+RAG demonstrates the fewest hallucinations while maintaining a comparatively good comprehensiveness score. We found that the tradeoff with comprehensiveness is partly thanks to its tendency to abstain from answering in certain questions, e.g., responding *"I'm unable to help, and don't have the ability to process and understand that."* (see *%resp* column in Table 4), which may be desired over misinformation in a patient-facing application. Computing the metrics only over the answered questions, this model receives a *Comp* score of 65.5% and a *Hall* score of 16.1, which is still lower than all other models, with the second-highest comprehensiveness score. However, Bing Chat and BARD, which also abstain, appear to underperform compared to their base models. This discrepancy might stem from our prompts lacking task optimization and their generic web retrieval, especially failing to focus on consumer health inquiries in the medical domain from reliable sources.

MedAlpaca performs poorly on K-QA. Even though MedAlpaca was fine-tuned specifically for the biomedical domain and intended for use as medical conversational AI, it exhibites the poorest results on both metrics, with an especially high hallucination rate. These findings indicate a mismatch between closed-QA (e.g., medical exams and short answers) and real-world patient questions which require the generation of long medical answers.

6 Conclusion

We introduce K-QA, a question-answering benchmark with real-world patients' questions and carefully curated physician answers. We formulate metrics that quantify how well a predicted answer covers important information and to what extent it contradicts gold answers. LLMs improve with size and augmented generation, but there is still a lot of room for improvement in both comprehensiveness and hallucination rate.

Limitations

One of the major limitations of our evaluation approach is its reliance on LLMs for approximating the entailment relation between ground-truth and predicted answers. In addition, the model which is used for the evaluation (GPT-4) is also then tested on the QA task, which may further confound our findings. While this was done in various recent works, it may propagate noise into the evaluation process, and yield a costly evaluation protocol. To mitigate this concern, we measure the agreement between human annotators and predicted labels, finding overall good agreement, while reducing evaluation costs an important avenue for future work (Perlitz et al., 2023). For the closed models, GPT-3.5, GPT-4, PALM-2, BARD and Bing, the responses are based on API calls, which are subject to changes in model versions, making reproducibility difficult.

Ethics Statement

The data in K-QA originates from deidentified real-world patient conversations that have been manually reviewed to ensure there it contains no personal information and revolves around general medical questions. The answers in K-QA were manually written by medical doctors, who did not use any automated writing assistance and wrote their answers with a general audience in mind. Our legal team has reviewed and approved the methodology used.

7 Acknowledgments

We would like to express our gratitude to Carmel Vider, Lee Herzog, Tamar Brufman, Nadia Mordenfeld, Lior Hayat, Ohad Simons, Jessica Thien, Shira Ben Porat, Itay Omer, Assaf Frajman, and Noa Bleistei n for curating and reviewing the medical answers. In addition, we would like to thank Nadav Rabani and Ya'ara Arkin for insightful discussion that influenced the direction of this work.

References

Asma Ben Abacha, Eugene Agichtein, Yuval Pinter, and Dina Demner-Fushman. 2017. Overview of the medical question answering task at trec 2017 liveqa. In *TREC*, pages 1–12.

Roee Aharoni, Shashi Narayan, Joshua Maynez, Jonathan Herzig, Elizabeth Clark, and Mirella Lapata. 2023. Multilingual summarization with factual consistency evaluation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3562–3591, Toronto, Canada. Association for Computational Linguistics.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Asma Ben Abacha, Eugene Agichtein, Yuval Pinter, and Dina Demner-Fushman. 2017. Overview of the medical question answering task at trec 2017 liveqa. In *TREC 2017*.

Asma Ben Abacha, Yassine Mrabet, Mark Sharp, Travis Goodwin, Sonya E. Shooshan, and Dina Demner-Fushman. 2019. Bridging the gap between consumers' medication questions and trusted answers. In *MEDINFO 2019*.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Conference on Empirical Methods in Natural Language Processing*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

John J Cherian, Isaac Gibbs, and Emmanuel J Candès. 2024. Large language model validity via enhanced conformal prediction methods. *arXiv preprint arXiv:2406.09714*.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*.

Dina Demner-Fushman, Yassine Mrabet, and Asma Ben Abacha. 2020. Consumer health information and question answering: helping consumers find answers to their health-related information needs. *Journal of the American Medical Informatics Association*, 27(2):194–201.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Sebastian Gehrmann, Elizabeth Clark, and Thibault Sellam. 2023. Repairing the cracked foundation: A survey of obstacles in evaluation practices for generated text. *Journal of Artificial Intelligence Research*, 77:103–166.

Tianyu Han, Lisa C Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K Bressem. 2023. Medalpaca–an open-source collection of medical conversational ai models and training data. *arXiv preprint arXiv:2304.08247*.

Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. 2021. q^2: Evaluating factual consistency in knowledge-grounded dialogues via question generation and question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7856–7870, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2021. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William W Cohen, and Xinghua Lu. 2019. Pubmedqa: A dataset for biomedical research question answering. *arXiv preprint arXiv:1909.06146*.

Qiao Jin, Zheng Yuan, Guangzhi Xiong, Qianlan Yu, Huaiyuan Ying, Chuanqi Tan, Mosha Chen, Songfang Huang, Xiaozhong Liu, and Sheng Yu. 2022. Biomedical question answering: a survey of approaches and challenges. *ACM Computing Surveys (CSUR)*, 55(2):1–36.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Evelyn Kai-Yan Liu. 2022. Low-resource neural machine translation: A case study of Cantonese. In

Proceedings of the Ninth Workshop on NLP for Similar Languages, Varieties and Dialects, pages 28–40, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *arXiv preprint arXiv:2305.14251*.

Moran Mizrahi, Guy Kaplan, Dan Malkin, Rotem Dror, Dafna Shahaf, and Gabriel Stanovsky. 2024. State of what art? a call for multi-prompt llm evaluation.

Ani Nenkova, Rebecca Passonneau, and Kathleen McKeown. 2007. The pyramid method: Incorporating human content selection variation in summarization evaluation. *ACM Transactions on Speech and Language Processing (TSLP)*, 4(2):4–es.

OpenAI. 2023. Gpt-4 technical report.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pages 248–260. PMLR.

Yotam Perlitz, Elron Bandel, Ariel Gera, Ofir Arviv, Liat Ein-Dor, Eyal Shnarch, Noam Slonim, Michal Shmueli-Scheuer, and Leshem Choshen. 2023. Efficient benchmarking (of language models). *ArXiv*, abs/2308.11696.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. 2022. Measuring and narrowing the compositionality gap in language models. *ArXiv*, abs/2210.03350.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2023. Gpqa: A graduate-level google-proof q&a benchmark. *ArXiv*, abs/2311.12022.

Max Savery, Asma Ben Abacha, Soumya Gayen, and Dina Demner-Fushman. 2020. Question-driven summarization of answers to consumer health questions. *Scientific Data*, 7(1):322.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2023. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. 2023. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. 2015. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16(1):1–28.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

WHO. 1993. *The ICD-10 classification of mental and behavioural disorders: diagnostic criteria for research*, volume 2. World Health Organization.

Xiao Zhang, Ji Wu, Zhiyang He, Xien Liu, and Ying Su. 2018. Medical exam question answering with large-scale reading comprehension. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

A General Scoring Framework

In order to expand upon the definitions provided in 4, we define $w \in \mathcal{W}$ as the weight assigned to a specific statement $s \in \mathcal{S}$ for a given question. Additionally, we consider an NLI model $\mathcal{N}(premise, hypothesis)$ designed to classify each pair of *(premise, hypothesis)* into one of three labels: $l \in \{entail, contradict, neutral\}$. Within this framework, we view the hypothesis as the statement, and the generated response ($\hat{P}$) as the premise, denoted as $\mathcal{N}(\hat{P}, s)$. We define $f(\mathcal{N}(\hat{P}, s), l)$ as a function that takes the output of the NLI model and a predefined logical relation, such as "does not contradict," and returns a boolean value. The formula representing this process is as follows:

$$\sum_{s \in \mathcal{S}} w(s) \cdot f(\mathcal{N}(\hat{P}, s), l)$$

This mathematical expression quantifies how well the generated response aligns with a predefined logical condition, while taking into account the weights assigned to individual statement. The formulation of this equation is aligned with the metrics presented at section 4. For *Hall* computation, $w(s)$ is set to 1, whereas for *Comp*, $w(s)$ takes the value of 1 if s is in *Must Have* and 0 if s falls within *Nice to Have*.

B Annotators' Guidelines

B.1 Decomposition to Self-Contained Statements

We aim to evaluate the medical accuracy of responses generated by language models, specifically concerning stand-alone questions within medical conversations initiated by users. Given the potential for these answers to be open-ended, the evaluation task presents inherent challenges. To address this, we've devised a set of guidelines that break down answers into two components: "Must-have" statements, deemed essential for inclusion, and "Nice-to-have" statements, which, while beneficial, are not obligatory. Our objective is to formulate statements that are concise, accurate, definitive, and self-contained. It is imperative to ensure that the curated statements are medically correct and logically well-structured. The evaluation process is conducted independently for each statement, emphasizing the importance of avoiding overlap between statements to maintain clarity.

Guidelines for Various Scenarios

Below, a specific explanation is provided for different scenarios, accompanied by examples of good and bad options for decomposing an answer.

- **List of Unrelated Crucial Entities:** If the answer comprises a list of entities (e.g., red flags, vaccines, symptoms), and each entity is independently significant, consider treating each entity as a separate and distinct statement. However, if the list of entities is not critical to include in its entirety (e.g., suggestions for weekly menu options), these entities should be combined into a single statement and categorized as *Nice to Have*. The validation for these statements should ensure non-contradiction with the physician's input.

Question	***Must Have* - Good**	***Must Have* - Bad**
I am a young healthy adult, flying to Brazil next month. What vaccination should I take?	-Vaccination for Yellow Fever is recommended before traveling to Brazil. -Vaccination for Typhoid is recommended before traveling to Brazil.	Since you are traveling to Brazil next month, it is recommended to get fully vaccinated for several vaccines, including Typhoid and yellow fever.

Why is this example considered suboptimal?

- Single Answer Instead of Two: Instead of presenting two separate statements, the response combines both vaccines into a single answer. To enhance clarity and evaluation, it is recommended to break down such responses into distinct and independent statements, especially when the mentioned vaccines are not interdependent.

- Excessive Length: The response is too long and contains unnecessary information, particularly with the inclusion of prefixes like "Since you. . . ". The focus should be on keeping the information concise and relevant.

- Overly Specific: The mention of the timeframe "next month" is overly specific and potentially misleading. It is crucial to provide information that is essential and directly related to the question.

- **AND/OR statements:** When entities have a logical relationship (e.g., treatment options), express them in statements following their logical connection rather than separating them. Entity 1 AND/OR Entity 2.

Question	***Must Have* - Good**	***Must Have* - Bad**
What is the best hypertension treatment for patients who are also diabetic?	Either angiotensin-converting enzyme (ACE) inhibitors OR angiotensin receptor blockers (ARBs), but not both.	-Angiotensin-converting enzyme (ACE) inhibitors -Angiotensin receptor blockers (ARBs)
	***Nice to Have* - Good**	***Nice to Have* - Bad**
	-angiotensin receptor blockers (ARBs) can be taken alone or with the following medications (thiazide and/or ccb). -angiotensin-converting enzyme (ACE) can be taken alone or with the following medications (thiazide and/or ccb)	

Why is this example considered suboptimal?

- Misleading statements: The entities ACE and the ARB are dependent on each other. If the language model (LLM) provides an answer recommending the patient take both ACE and ARB, it would be medically incorrect but might receive a high score in our evaluation method. In such cases, as ACE and ARB are distinct treatment options, we need to combine them in a statement to emphasize that only one of them can be prescribed, not both.

- Lack of inclusivity: Besides ACEs and ARBs, the treatment plan may involve other medications. We want to ensure that the response does not contradict the LLM's answer.

Moreover, if there are additional medications that can be prescribed alongside ACE or ARB, they should be listed in parentheses next to thiazide and/or CCB.

- **IF Statements:** Similar to AND/OR statements, maintain the coherence of "IF statements" if the cause is not a stand-alone factor.

	***Nice to Have* - Good**	***Nice to Have* - Bad**
What is the best hypertension treatment for patients who are also diabetic?	If lifestyle modifications alone are not effective in reducing blood pressure, medications may be necessary;	-Medications are needed to reduce the blood pressure. -Sometimes life modifications are not enough to reduce the blood pressure.

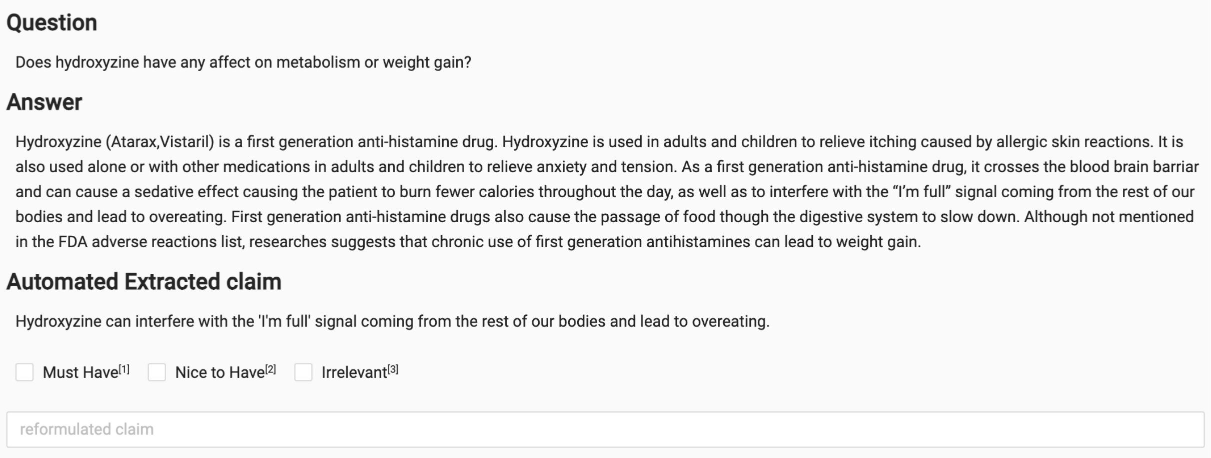
Why is this example considered suboptimal?

- Misleading statements: Splitting the IF statement leads to misleading statements. Even if the "bad examples" are medically correct, we might deviate from the intended verification of the IF statement.

- **Drugs Inclusion:** Include both the family drug name (generic) and trade name when dealing with drugs. For instance, "Low-risk drugs during pregnancy include aminosalicylates, such as sulfasalazine (Azulfidine) and mesalamine (Asacol, Pentasa).

C Annotators' Interface

Figures 6 display the annotation interface used for human evaluation during dataset creation. In this interface, annotators executed the second and third steps (described in 3.2 & 3.2). They were presented with the patient question, the free-form answer written by the physician, and the suggested decomposition of statements provided by GPT-4. Annotators were tasked with confirming, modifying, or removing the suggested decomposition to ensure relevance and self-containment. Additionally, they categorized the statement into one of the categories: *Must Have*, *Nice to Have*, or deemed it irrelevant.



Question

Does hydroxyzine have any affect on metabolism or weight gain?

Answer

Hydroxyzine (Atarax,Vistaril) is a first generation anti-histamine drug. Hydroxyzine is used in adults and children to relieve itching caused by allergic skin reactions. It is also used alone or with other medications in adults and children to relieve anxiety and tension. As a first generation anti-histamine drug, it crosses the blood brain barriar and can cause a sedative effect causing the patient to burn fewer calories throughout the day, as well as to interfere with the “I’m full” signal coming from the rest of our bodies and lead to overeating. First generation anti-histamine drugs also cause the passage of food though the digestive system to slow down. Although not mentioned in the FDA adverse reactions list, researches suggests that chronic use of first generation antihistamines can lead to weight gain.

Automated Extracted claim

Hydroxyzine can interfere with the 'I'm full' signal coming from the rest of our bodies and lead to overeating.

□ Must Have[1] □ Nice to Have[2] □ Irrelevant[3]

reformulated claim

Figure 6: An example of the annotator’s interface.

D Prompt Templates

D.1 Decomposition Free-Form to Statements

```
# OVERALL INSTRUCTIONS
You are an expert in understanding logical relationships. This is a Semantic Content Unit (SCU) extraction task. Given a pair of
Question and Answer, your goal is to create a list of self-contained and concise claims. Each claim should be able to stand alone
and be independent of other claims. Your claims should encompass all the information present in the answer.

# TASK INSTRUCTIONS
- List of Possible Causes: For scenarios involving multiple entities like red flags, vaccines, symptoms, etc., generate separate
claims for each entity. This increases the number of claims.
- OR Claims: When medical entities are presented in an "OR" context, treat them as distinct claims.
- IF Claims: When an "if statement" is present, preserve the "if statement" context while creating the claim.
- XOR Claims: When entities have an XOR logical relationship (e.g., treatment options), create separate claims for each option.

# EXAMPLE CLAIM FORMAT - List Format: "Possible cause for [CONDITION] in [DEMOGRAPHIC] can be [ENTITY]."
- OR Format: "Possible causes include: [ENTITY X], [ENTITY Y], and [ENTITY Z]."
- OR Format: "The [CONTEXT] of treatments such as [TREATMENT X], [TREATMENT Y], and [TREATMENT Z], is not well established." - IF
Format: "[CONTEXT], please seek medical attention if [CONDITIONS]."
- XOR Format: "Either take [TREATMENT X] or [TREATMENT Y], but not both."
--
{format_instructions}
--

# TASK EXAMPLE
Question: I am a 33-year-old female with right lower abdominal pain, what could it be? Answer: Possible causes for right lower
abdominal pain in a young female are Appendicitis, Inflammatory bowel disease, Diverticulitis, Kidney stone, urinary tract infection,
Ovarian cyst or torsion, Ectopic pregnancy, Pelvic inflammatory disease, endometriosis. Please seek medical attention if the pain
is sudden and severe, does not go away, or gets worse, is accompanied by fever, nausea and vomiting, or if you have noticed blood
in urine or in stool.
Claims: [ Possible cause for right lower abdominal pain in a young female: Appendicitis, Possible cause for right lower abdominal
pain in a young female: Ovarian cyst or torsion, Possible cause for right lower abdominal pain in a young female: Ectopic pregnancy,
Possible cause for right lower abdominal pain in a young female: Pelvic inflammatory disease, Possible cause for right lower
abdominal pain in a young female: Kidney stone, Possible cause for right lower abdominal pain in a young female: Urinary tract
infection, Possible cause for right lower abdominal pain in a young female: Diverticulitis, Possible cause for right lower abdominal
pain in a young female: Inflammatory bowel disease, Possible cause for right lower abdominal pain in a young female: Endometriosis,
Please seek medical attention if the pain is sudden and severe, Please seek medical attention if the pain is accompanied by fever,
Please seek medical attention if the pain is accompanied by nausea and vomiting, Please seek medical attention if the pain is
accompanied by blood in urine, Please seek medical attention if the pain is accompanied by blood in stool, Possible cause for right
lower abdominal pain in a young female: Emotional stress ]

# TASK EXAMPLE
Question: So what does the non reactive mean for the hep a igm Answer: Hep A IgM refers to a specific type of antibody called
Immunoglobulin M (IgM) against the virus hepatitis A. When infected with hepatitis A, these antibodies are detectable at symptom
onset and remain detectable for approximately three to six months. These antibodies might also be detectable in the first month
after hepatitis A vaccination. A negative or non-reactive result means no IgM antibodies against hepatitis A found in your serum,
meaning the absence of an acute or recent hepatitis A virus infection.
Claims: [ A negative or non-reactive result means that there were no IgM antibodies against hepatitis A found in your serum, The
absence of IgM antibodies against hepatitis A in your serum indicates the absence of an acute or recent hepatitis A virus infection,
Hep A IgM refers to a specific type of antibodies called Immunoglobulin M (IgM) against the virus hepatitis A, These antibodies
might also be detectable in the first month after hepatitis A vaccination, These antibodies remain detectable for approximately
three to six months after infection, When infected with hepatitis A, these antibodies are detectable at the time of symptom onset ]

# TASK EXAMPLE
Question: What medications are contraindicated for a pregnant woman with ulcerative colitis? Answer: methotrexate (Otrexup, Rasuvo,
RediTrex) and thalidomide (Contergan, Thalomid) are both considered contraindicated for treatment of UC in pregnancy. possible
treatment for UC during pregnancy include low-risk drugs such as aminosalicylates (sulfasalazine and mesalamine), immunomodulators
(azathioprine, cyclosporine A ,6-mercaptopurine) and corticosteroids. Biological agents such as Infliximabl, Adalimumab, Vedolizumab
and Ustekinumab is generally avoided during pregnancy as their safety in pregnancy is not well established yet.
Claims: [ Methotrexate (Otrexup, Rasuvo, RediTrex) is contraindicated for treatment of ulcerative colitis in pregnancy, Thalidomide
(Contergan, Thalomid) is contraindicated for treatment of ulcerative colitis in pregnancy, Aminosalicylates (sulfasalazine and
mesalamine) are considered low-risk drugs for treatment of ulcerative colitis during pregnancy, Immunomodulators (azathioprine,
cyclosporine A, 6-mercaptopurine) are considered low-risk drugs for treatment of ulcerative colitis during pregnancy, Corticosteroids
are considered low-risk drugs for treatment of ulcerative colitis during pregnancy, Treatment for ulcerative colitis during pregnancy
with biological agents such as Adalimumab is generally avoided during pregnancy as their safety in pregnancy is not well established
yet, Treatment for ulcerative colitis during pregnancy with biological agents such as Vedolizumab is generally avoided during
pregnancy as their safety in pregnancy is not well established yet, Treatment for ulcerative colitis during pregnancy with biological
agents such as Infliximab is generally avoided during pregnancy as their safety in pregnancy is not well established yet, Treatment
for ulcerative colitis during pregnancy with biological agents such as Ustekinumab is generally avoided during pregnancy as their
safety in pregnancy is not well established yet, ]

# YOUR TASK
Question: {question}
Answer: {answer}
Claims:
```

D.2 Automatic Evaluation

```
# OVERALL INSTRUCTIONS
- You have a deep understanding of logical relationships, such as entailment and contradiction, to evaluate given triplets of (question, premise, hypothesis).

# TASK INSTRUCTIONS
Your goal is to determine whether the Premise effectively contradicts the corresponding Hypothesis. Carefully analyze each triplet, focusing on details. - If the premise and the hypothesis are unrelated or lack sufficient evidence to ascertain their truthfulness, label your answer as False. - be vigilant in identifying cases where the premise doesn't rule out the possibility of an entity (e.g., vaccine, symptom) appearing in the hypothesis. In such cases, classify the answer as False. - Approach each question methodically, considering the step-by-step process outlined below.

# INPUT DATA
Question: What does trich test for? Let's think step by step.
Premise: The term "trich test" can refer to two different medical tests, depending on the context. Here are the two possibilities: Trichomoniasis Test: Trichomoniasis is a sexually transmitted infection (STI) caused by the parasite Trichomonas vaginalis. The trichomoniasis test, also known as a trich test or trichomonas test, is used to detect the presence of this parasite in the body. The test is typically performed on a sample of vaginal discharge in women or urine in men. Trichogram: A trichogram is a diagnostic test used to evaluate hair loss and assess the health and condition of hair follicles. It involves plucking a small number of hairs from the scalp and examining them under a microscope. It's important to note that without additional context, it's difficult to determine which specific test you are referring to. Hypothesis: Trichamoniasis- a parasitic infection that can cause your symptoms.
Answer: According to the premise "trich test" refer to two different medical tests. A Trichamoniasis test is one of them, which is used to detect this parasite's presence. As stated in the hypothesis, the trich test is used to diagnose parasitic infections. Ths premise entails the hypothesis. The answer is False.
###
Question: Can diabetics eat sweets? Let's think step by step.
Premise: Individuals with diabetes are recommended to limit their consumption of sweets to one or two times per week. It is also suggested being selective with desserts and to focus on foods with a low glycemic index, such as high fiber foods like whole grains and legumes, as well as certain lower sugar fruits like berries, melons, and apples.
Hypothesis: It is recommended that diabetics avoid sweets.
Answer: The premise suggests that diabetics can eat sweets but limit their consumption. According to the hypothesis diabetics should avoid sweets. Diabetics are allowed to consume sweets according to the premise, but they are prohibited according to the hypothesis. There is a contradiction between the premise and the hypothesis. The answer is True.
###
Question: 25 yo female with right lower abdominal pain, what might be causing it? Let's think step by step.
Premise: Right lower abdominal pain in a 25-year-old female could be caused by a variety of medical conditions.
Some potential causes include: Ovarian cyst: a fluid-filled sac on the ovary - Ectopic pregnancy: a pregnancy that occurs outside the uterus.
Hypothesis: possible cause for right lower abdominal pain in a young female can be Appendicitis.
Answer: The premise lists several potential causes of right lower abdominal pain in a 25-year-old female, not including appendicitis. The hypothesis states that Appendicitis could be a cause of right lower abdominal pain in a young female. There is no direct contradiction between the premise and the hypothesis, as the premise does not exclude the possibility of appendicitis as the cause of the pain. The answer is False.
###
Question: Can a headache last longer than a few days? Let's think step by step.
Premise: Yes, it is possible. If you are experiencing a headache that lasts longer than a few days, it is important to see a doctor to get the appropriate treatment.
This will help to relieve the pain and prevent any further complications.
Hypothesis: It is not a cause for concern if a headache lasts longer than a few days.
Answer: This premise acknowledges that a headache can last for several days, but emphasizes that seeing a doctor to prevent further complications is important. According to this hypothesis, headaches lasting longer than a few days are not cause of concern. There is a contradiction between the premise and hypothesis due to the premise emphasizing the importance of seeking medical consultation, while the hypothesis posits that there is no cause for concern. The answer is True.
###

Question: {question} Let's think step by step.
Premise: {llm_answer}
Hypothesis: {physician_answer}
Answer:
```

D.3 Question-Answering

vanilla prompt

```
Question: {question}
Answer:
```

vanilla prompt for RAG

```
# TASK:
Answering question.

# INSTRUCTIONS:
- Use ONLY our verified sources to answer the question.
- If the answer is not in our sources, answer ONLY "I don’t know".

# VERIFIED SOURCES
Our verified sources: {context}

# INPUT
Question: {question}

# OUTPUT
Answer:
```

engineered prompt

```
# TASK:
You are a medical expert and your task is to write a helpful human-like text and answer to the point.

# INSTRUCTIONS:
- Identify and address the patient’s focus of interest, but include relevant information even if the patient didn’t directly ask it.
- If relevant, provide information about specific cases, including children, pregnant women, allergies, or chronic conditions.
- If you don’t know the answer, answer ONLY ’I don’t know’.
- If the answer contains enumeration of factors/symptoms/conditions use bullet points to organize the information.

# EXAMPLES:
Question: Is it safe to take Macrobid?
Answer: Macrobid is generally considered safe to take when prescribed by a doctor and used according to their instructions.
Rationale: The answer is good, but it is missing important information regarding special cases, such as pregnant women, kids and
comorbidities.
A better answer would include that Macrobid is not recommended for use in the last 2 to 4 weeks of pregnancy
###
Question: Ok Do I need any antibiotics or something for yeast infection?
Answer: Over-the-counter antifungal creams or suppositories applied internally to the vagina can effectively treat vaginal yeast
infections, while a single oral dose of prescription fluconazole (Diflucan) may also be an option.
Rationale: the answer is good however it does not include relevant explanation about the condition such as Yeast infections are
commonly caused by fungi, particularly Candida albicans, which is a natural inhabitant of the body but can cause infections in
different areas such as the skin, mouth, and vagina. In addition, it assumes the patient is asking about vaginal yeast infection
and excludes information about other possible infections caused by yeast.
###
Question: the vomiting, tight abdomen pain and burning pain may just be result of pcos?
Vomiting and tight abdominal pain are not common clinical presentations of PCOS. Vomiting and abdominal pain are nonspecific symptoms
which could be an indication for a variety of illnesses. In case of this presentation, you should be evaluated by a physician, and
treated accordingly.
Rationale: the answer is good, but it does not emphasize enough that the patient’s symptoms are most likely caused by something
else and what is the differential diagnosis in this case.

# INPUT
Question: {question}
Answer:
```

engineered prompt for RAG

```
# TASK:
You are a medical expert whose task is to provide a helpful, concise, human-like response to a patient’s question based on verified medical sources.

# INSTRUCTIONS:
- Use our verified sources to answer the question at the end.
- If the answer is not in our sources, answer ONLY ’I don’t know’.
- It is important to assume that there may be a medical concern underlying the patient’s questions.
- If relevant, provide information about specific cases, including children, pregnant women, allergies, or chronic conditions.
- If the answer contains enumeration of factors/symptoms/conditions use bullet points to organize the information.
- Avoid repetition of details.

# EXAMPLES
Question: Is it safe to take Macrobid?
Answer: Macrobid is generally considered safe to take when prescribed by a doctor and used according to their instructions.
Rationale: The answer is good, but it is missing important information regarding special cases, such as pregnant women, kids and comorbidities.
A better answer would include that Macrobid is not recommended for use in the last 2 to 4 weeks of pregnancy
###
Question: I had my gallbladder removed in 2015, so that rules out gallstones, right?
Answer: Yes, as gallstones are stones that form inside the gallbladder, if the gallbladder is removed, gallstones cannot be the cause of future symptoms.
Rationale: The answer is correct however it does not provide the patient with an important clinical tie as even without a gallbladder, stones can develop anywhere in the biliary system and cause similar symptoms to gallstone disease.
###
Question: the vomiting, tight abdomen pain and burning pain may just be result of pcos?
Answer: Polycystic ovary syndrome (PCOS) is an endocrine (hormonal) insufficiency which is characterized by polycystic ovaries and has a variety of effects including anovulation and irregular menstrual cycles which in turn could causes fertility issues, hyperandrogenism (high levels of androgens causing coarse body hair growth in a male pattern and acne) and insulin resistance causing type 2 diabetes mellitus, obesity, and hypertension.
Vomiting and tight abdominal pain are not common clinical presentations of PCOS. Vomiting and abdominal pain are nonspecific symptoms which could be an indication for a variety of illnesses. In case of this presentation, you should be evaluated by a physician, and treated accordingly.
Rationale: The answer is good, but it does not emphasize enough that the patient’s symptoms are most likely caused by something else and what is the differential diagnosis in this case.

# CONTEXT
Our verified sources: {context}

# INPUT
Question: {question}
Answer:
```

Large Language Models for Biomedical Knowledge Graph Construction: Information extraction from EMR notes

Vahan Arsenyan[1], **Spartak Bughdaryan**[1], **Fadi Shaya**[2], **Kent Small**[3],
Davit Shahnazaryan[1,2],
[1]Yerevan State University, [2]Amaros AI, [3]Macula and Retina Institute

Abstract

The automatic construction of knowledge graphs (KGs) is an important research area in medicine, with far-reaching applications spanning drug discovery and clinical trial design. These applications hinge on the accurate identification of interactions among medical and biological entities. In this study, we propose an end-to-end machine learning solution based on large language models (LLMs) that utilize electronic medical record notes to construct KGs. The entities used in the KG construction process are diseases, factors, treatments, as well as manifestations that coexist with the patient while experiencing the disease. Given the critical need for high-quality performance in medical applications, we embark on a comprehensive assessment of 12 LLMs of various architectures, evaluating their performance and safety attributes. To gauge the quantitative efficacy of our approach by assessing both precision and recall, we manually annotate a dataset provided by the Macula and Retina Institute. We also assess the qualitative performance of LLMs, such as the ability to generate structured outputs or the tendency to hallucinate. The results illustrate that in contrast to encoder-only and encoder-decoder, decoder-only LLMs require further investigation. Additionally, we provide guided prompt design to utilize such LLMs. The application of the proposed methodology is demonstrated on age-related macular degeneration.

Data and Code Availability The dataset utilized in this study is provided by the Macula and Retina Institute and is not accessible to the public.

Institutional Review Board (IRB) This research does not require IRB approval.

1 Introduction

There are several biomedical data corpora available that provide valuable knowledge, and one such source is PubMed (Kilicoglu et al., 2012). PubMed is a search engine that accesses MEDLINE (Kilicoglu et al., 2012), which is a database of abstracts of medical publications and references. Moreover, the widespread adoption of electronic medical records (EMR) has brought various opportunities for medical knowledge discovery. Knowledge graphs (KG) are often used for knowledge discovery, because graph-based abstraction offers numerous benefits when compared with traditional representations. They have been applied to various areas of healthcare, including identifying protein functions (Santos et al., 2022), drug repurposing (Drancé et al., 2021), and detecting healthcare misinformation (Cui et al., 2020). Another application may be a clinical trial design (Skelly et al., 2012), during which identification of confounding variables is an important step. Confounding variables may mask an actual association, or, more commonly falsely demonstrate an apparent association between the treatment and outcome when no real association between them exists.

KGs are a powerful tool for organizing and representing knowledge in a graph structure, where nodes represent entities within a specific domain, while edges symbolize relationships between these entities. The type of relationships may vary depending on the domain, allowing for the use of directed or undirected graphs. For example, in (Nordon et al., 2019), they employed a directed graph to encode causal relationships between diseases. Other KGs may utilize both symmetric and asymmetric relationships. In our work, we specifically focus on using directed graphs to represent relationships between diseases and various factors, treatments, and manifestations that coexist with a patient while experiencing the disease (referred to as 'coexists_with').

Recent advancements in large language models (LLM) offer an opportunity to think about their ability to learn valuable representations from the knowl-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 295–317
August 16, 2024. ©2024 Association for Computational Linguistics

edge encoded in medical corpora. Effectively analyzing textual data and KG construction requires extensive domain knowledge and is often a time-consuming process for medical experts. To address this challenge, we propose an end-to-end method for automatically constructing knowledge graphs from electronic medical record (EMR) notes using LLMs, specifically through relation extraction.

Previous studies have suggested the utilization of specific LLMs for clinical relation extraction (Agrawal et al., 2022; Sushil et al., 2022). However, due to the inherent safety-critical nature of healthcare, we conducted a comprehensive analysis of the performance and safety attributes of LLMs with varying architectures. To evaluate and assess their potential for medical applications and to address potential safety concerns, we introduced a manually annotated, private dataset and benchmarked the performance of 12 distinct LLMs. We have not performed an analysis on publicly available EMR datasets, such as MIMIC-III (Johnson et al., 2016), because some of the models have used these datasets for training or fine-tuning. Our analysis revealed that in contrast with encoder-only and encoder-decoder models, decoder-only models need further guidance to output in a structured manner, which is required for relation extraction to construct the KG. We, therefore, introduced a guided prompt design that helped to utilize some of such LLMs for our task and analyzed issues that are making others unsuitable. This rigorous assessment forms a critical foundation for the safe and effective deployment of LLMs in the healthcare domain. Our work takes the form of the following contributions:

- We present a end-to-end method leveraging LLMs for the automatic construction of KGs from EMR notes
- We conduct an extensive and rigorous evaluation of the performance of 12 LLMs of various architectures specifically tailored for clinical relation extraction
- We provide guided prompt design to utilize decoder-only LLMs for relation extraction to construct KG between aforementioned medical entities

2 Related Work

One notable success in the construction of knowledge bases (KBs) from biomedical textual data is SemRep (Rindflesch and Fiszman, 2003). SemRep is a rule-based system that combines syntax and semantics with biomedical domain knowledge contained in the Unified Medical Language System (UMLS) (Bodenreider, 2004) for semantic relation extraction. The range of predicates in SemRep is diverse, including molecular interactions, disease etiology, and static relations. Shalit et al. (Nordon et al., 2019) further improve the precision of SemRep by adding three additional filtration steps.

As one may observe, SemRep utilizes various levels of language modeling. It has been experimentally demonstrated that LLMs intrinsically learn these levels of language specification, without explicit programming (Søgaard, 2021). In (Sung et al., 2021), BERT-based models with probing are used to extract relations between biomedical entities. The authors observe that, although LLMs can extract biomedical knowledge, they are biased towards frequently occurring entities present in prompts. We do not argue about the bias of LLMs, but rather the complexity of extracting relations via probing. We propose providing larger context information than that which is solely present in the prompt.

(Rotmensch et al., 2017) utilizes both structured and unstructured data from EMR to construct knowledge graphs. The structured data includes ICD-9 (International Classification of Diseases) diagnosis codes, while the unstructured data comprises various notes written by physicians and nurses to track a patient's course. On the other hand, (Chandak et al., 2023) employs 20 multimodal data resources to describe a disease with various relationships representing different biological scales. However, in this work, we solely concentrate on clinical notes for information extraction and KG construction.

(Trajanoska et al., 2023) makes connection between LLMs and semantic reasoning to automatically generate a KG on the topic of sustainability. It further populates it with concrete instances using news articles from the internet. It experiments with REBEL (Huguet Cabot and Navigli, 2021) and ChatGPT and shows that ChatGPT (OpenAI, 2023) is able to automatically create KGs from unstructured text, if reinforced with detailed instructions.

The paper on few-shot clinical extraction using LLMs (Agrawal et al., 2022) discusses the challenge of extracting important variables from clinical data and presents an approach that leverages large language models, specifically InstructGPT

(Ouyang et al., 2022), for zero-shot and few-shot information extraction from clinical text. The authors demonstrate the effectiveness of this approach in several NLP tasks that require structured outputs, such as span identification, token-level sequence classification, and relation extraction. To evaluate the performance of the system, the authors introduce new datasets based on a manual reannotation of the CASI dataset (Moon et al., 2014).

We argue that our setup is more complex as we do not consider clean, well-written, academic corpora such as PubMed (Kilicoglu et al., 2012) and CASI (Moon et al., 2014). The EMR corpus contains a significant amount of grammatical errors ("there is some heme OD .. ?"). Practitioners use abbreviations and notations ("RTO") not defined in the context, obfuscating the underlying information even further. Our study benchmarks different LLMs of varying architectures and training procedures on this challenging dataset.

3 Dataset

For this cohort study, data was obtained from the EMR of the Macula & Retina Institute, an independent health system in Glendale, California, USA. The dataset included approximately 10,000 patient records of individuals with retina-related eye diseases who had visited the institute between 2008 and 2023. The study focused on extracting knowledge from the clinical notes, which are records of observations, plans, and other activities related to patient care. These notes contain a patient's medical history and reasoning and can be used to identify complex disease-related patterns such as potential treatments, causes, and symptoms. In total, the study analyzed 360,000 notes relating to 122 unique eye diseases.

3.1 Dataset preprocessing

Clinical notes often include repetitive segments following a standardized template used by medical practitioners, resulting in unnecessary computational overhead during the analysis. To address this issue, cosine similarity is computed between the embeddings of notes generated by Sentence T5 XXL (Ni et al., 2022). If the similarity score exceeds the threshold (referred to as threshold_preprocessing, detailed in Appendix F), priority is given to the note with a higher word count to retain more informative content. Additionally, notes containing fewer than 5 words are excluded from further analysis.

4 Proposed method

Our proposed method constructs a KG of diseases and their factors, treatments, and manifestations that the patient exhibits while undergoing the disease. To achieve this, the system initially identifies disease-specific notes as described in Subsection 4.1. Next, for each category of medical entity, we design set of questions (Subsection 4.5). We leverage an LLM to answer a pre-designed set of questions, taking into consideration the aforementioned disease-specific notes as contexts as described in Subsections 4.3 and 4.6. The list of LLMs that we experimented with are available in Subsection 4.2. All the experiments are performed on **8xV100** (32GB VRAM) GPUs which are widely accessible nowdays. The Subsection 4.7 discuss postprocessing techniques utilized to get the final relations to construct the KG.

4.1 Disease-specific notes identification

In clinical records, a single disease, denoted as d_{input}, may have multiple textual representations. The set of such expressions is denoted as D_{input}. These expressions may vary between clinics as well. To identify all instances of d_{input} in the records, we employ the Unified Medical Language System (UMLS) Metathesaurus (Bodenreider, 2004), a comprehensive repository of biomedical terminologies and ontologies containing over 3 million concepts and their corresponding aliases, such as diseases, drugs, and procedures. We first check if any of the expressions in $d_i \in D_{input}$ appear in the records within our dataset, and if so, we add the record to a list of disease-specific records for d_{input}. Sometimes, clinicians may make typographical errors when recording the condition in the notes. To account for this, we use the BioBERT NER model to extract a list of diseases, denoted as D_{note}, present in the record. We then calculate the cosine similarity between each expression $d_{note_i} \in D_{note}$ and $d_i \in D_{input}$. If the similarity is above threshold (denoted threshold_notes_identification, more in Appendix F) for at least one d_{note_i}, we add the record to the list of relevant notes for the disease d_i. Refer to Appendix C for more details on the algorithm.

4.2 Models

Table 1 shows all the models that we used in this paper. Our main objective revolves around exper-

Table 1: We show all the models used in this paper, as well as their size, architecture and the number of pretraining tokens. We focus only on pretraining data, and ignore any finetuning data. PTT stands for pretraining tokens.

Architecture	Model	Size	PTT
Encoder-only	BioBERT-SQuAD-v2	110M	137B
	BERT-SQuAD-v2	110M	137B
	RoBERTa-SQuAD-v2	125M	2.2T
Decoder-only	BioGPT	349M	-
	OPT	30B	180B
	OPT-IML-MAX	30B	180B
	Llama 2	70B	2T
	Vicuna	33B	2T
	BLOOM	176B	366B
	WizardLM	70B	2T
Encoder-decoder	FLAN-T5-XXL	11B	34B
	FLAN-UL2	20B	1T

imenting with various architectures of LLMs and analyzing their performance through a comprehensive evaluation that brings forward potential edge cases and safety attributes. To accomplish this, we conducted experiments using different LLM models categorized under three architectures: encoder-only, decoder-only, and encoder-decoder. Our next objective was to include as much diverse LLMs as possible encompassing variations in size as well as the number of pretraining tokens. For more detailed insights into each individual model, please refer to Appendix A.

4.3 Aligning LLMs for relation extraction

In this work, we assume only query access to a large language model (i.e., no gradients). The task is to identify relations by finding answers to specific queries. We explore two distinct approaches for aligning large language models to the task: open-book QA (Gholami and Noori, 2021) and in-context learning (Brown et al., 2020).

QA aims to find an answer to a given query. In open-book QA, a query comprises a question and a context. The system attempts to find an answer to the question from the context. It utilizes various variations of BERT (Devlin et al., 2019) language models, as described in Table 1. The model consists of two sets of dense layers with sigmoid activation in addition to the based BERT model. The first layer seeks the start of the answer sequences, while the second layer seeks the end of the answer sequences. For decoder-only and encoder-decoder models, we employ in-context learning (Brown et al., 2020), providing the LLM with a prompt consisting of a list of input-output pairs that answer a given query using the context. In this study, we focused on zero-shot (Wei et al., 2022), few-shot (Brown et al., 2020), and instruction-based prompting (Ye et al., 2023).

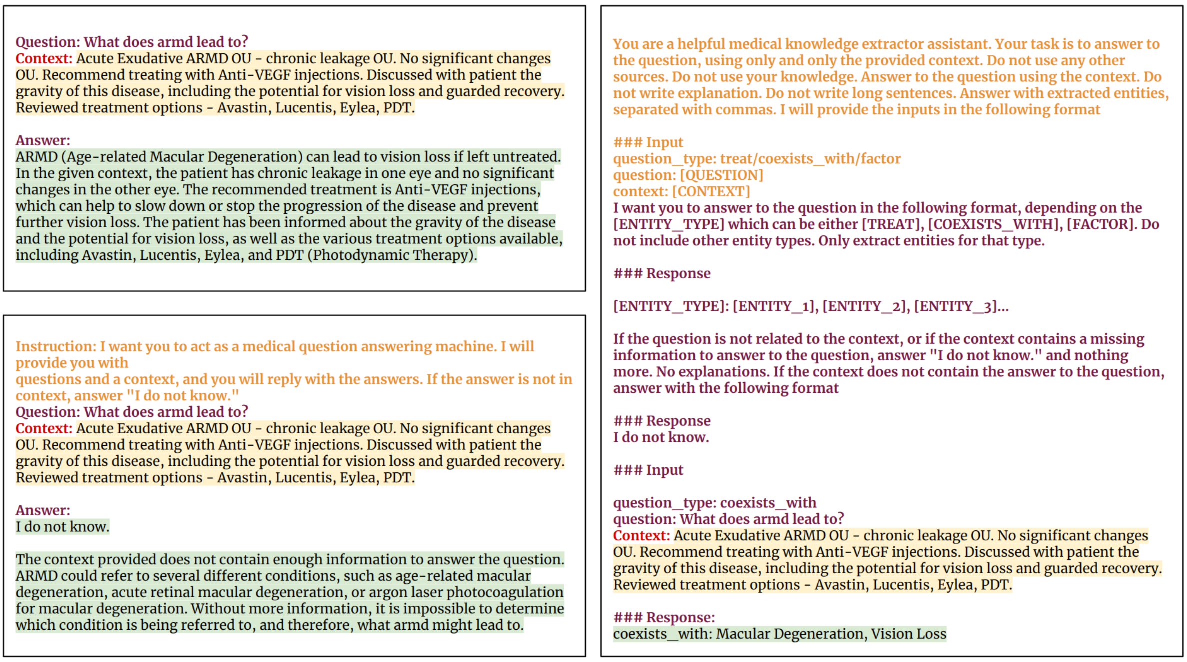
4.4 Prompt design

We follow a systematic and task-agnostic process to construct prompts as outlined in (Jimenez Gutierrez et al., 2022). As depicted in the examples in Figure 1, this method identifies three key components of a prompt: overall task instructions, a sentence introduction, and a retrieval message. In the case of zero-shot and few-shot approaches, simply entity-related questions are appended to the input (Figure 1 left-top). Additionally, for the few-shot approach, we provide an example input/output. For instruction-based prompting (Figure 1 left-bottom), overall task instructions are comprised of broad instructions for the task as it is described in (Jimenez Gutierrez et al., 2022).

Furthermore, we are introducing a prompt structure by defining a stringent input and response format. The primary focus is on extracting information exclusively from the provided context, accompanied by explicit instructions to incorporate specific entity types in the response. We have delineated a well-defined format for both the question and the response, promoting concise answers without explanations. Moreover, we have introduced a systematic approach to address situations where information is absent or questions are irrelevant, ensuring a consistent 'I do not know' response. In essence, these modifications contribute to enhancing the clarity and precision of the model's performance within this specific scientific context. For a comprehensive visualization of the refined prompt structure and its components, kindly consult the right block of Figure 1.

4.5 Question design

We define template questions like "What treats %s". The "%s" in the questions represents a placeholder for a disease. All the predicates (e.g. treats, affect, cause, factor) are taken from SemRep (Rindflesch and Fiszman, 2003). The questions are categorized into three types: treatment-related, factor-related, and coexists_with-related questions. The treatment-related questions inquire about methods



Question: What does armd lead to?
Context: Acute Exudative ARMD OU - chronic leakage OU. No significant changes OU. Recommend treating with Anti-VEGF injections. Discussed with patient the gravity of this disease, including the potential for vision loss and guarded recovery. Reviewed treatment options - Avastin, Lucentis, Eylea, PDT.

Answer:
ARMD (Age-related Macular Degeneration) can lead to vision loss if left untreated. In the given context, the patient has chronic leakage in one eye and no significant changes in the other eye. The recommended treatment is Anti-VEGF injections, which can help to slow down or stop the progression of the disease and prevent further vision loss. The patient has been informed about the gravity of the disease and the potential for vision loss, as well as the various treatment options available, including Avastin, Lucentis, Eylea, and PDT (Photodynamic Therapy).

Instruction: I want you to act as a medical question answering machine. I will provide you with
questions and a context, and you will reply with the answers. If the answer is not in context, answer "I do not know."
Question: What does armd lead to?
Context: Acute Exudative ARMD OU - chronic leakage OU. No significant changes OU. Recommend treating with Anti-VEGF injections. Discussed with patient the gravity of this disease, including the potential for vision loss and guarded recovery. Reviewed treatment options - Avastin, Lucentis, Eylea, PDT.

Answer:
I do not know.

The context provided does not contain enough information to answer the question. ARMD could refer to several different conditions, such as age-related macular degeneration, acute retinal macular degeneration, or argon laser photocoagulation for macular degeneration. Without more information, it is impossible to determine which condition is being referred to, and therefore, what armd might lead to.

You are a helpful medical knowledge extractor assistant. Your task is to answer to the question, using only and only the provided context. Do not use any other sources. Do not use your knowledge. Answer to the question using the context. Do not write explanation. Do not write long sentences. Answer with extracted entities, separated with commas. I will provide the inputs in the following format

Input
question_type: treat/coexists_with/factor
question: [QUESTION]
context: [CONTEXT]
I want you to answer to the question in the following format, depending on the [ENTITY_TYPE] which can be either [TREAT], [COEXISTS_WITH], [FACTOR]. Do not include other entity types. Only extract entities for that type.

Response

[ENTITY_TYPE]: [ENTITY_1], [ENTITY_2], [ENTITY_3]...

If the question is not related to the context, or if the context contains a missing information to answer to the question, answer "I do not know." and nothing more. No explanations. If the context does not contain the answer to the question, answer with the following format

Response
I do not know.

Input

question_type: coexists_with
question: What does armd lead to?
Context: Acute Exudative ARMD OU - chronic leakage OU. No significant changes OU. Recommend treating with Anti-VEGF injections. Discussed with patient the gravity of this disease, including the potential for vision loss and guarded recovery. Reviewed treatment options - Avastin, Lucentis, Eylea, PDT.

Response:
coexists_with: Macular Degeneration, Vision Loss

Figure 1: Each design element in the prompt is distinguished by a specific color annotation: orange represents overall task instructions, red indicates sentence introduction, purple signifies the retrieval message, and green is used for the LLM response. In the top-left corner, a basic prompt structure is outlined, which includes a sentence introduction and a retrieval message. The bottom-left section features an instruction prompt, encompassing overall task instructions as well. On the right, a newly introduced prompt structure is presented, encompassing all three components and incorporating input-output structure instructions.

to slow down the progression, decrease the chance, or reduce the risk of a specific condition. The factor-related questions aim to identify the causes, factors, or risks associated with a condition. The coexists_with-related questions explore any symptoms, effects, diseases, clinical tests, or behaviors that may manifest in the patient while experiencing the disease. The full list of questions for the LLM queries is available in Appendix B.

4.6 Relation extraction

We query an LLM for each disease $d \in D$ with a question $q(d)$ and a related context $c \in C(d)$ (refer to Appendix C for more details). The LLM returns a list of answers with their corresponding probabilities for each query quartet $\langle d, q(d), c, t \rangle$ where t identifies the question type, i.e. *treatment, factor, and coexists with*. As a single probability estimate may be unreliable (Nordon et al., 2019), we keep the relation triplet $\langle e, t, d \rangle$ if the LLM has returned e as an answer to any question of category t more than relation_occurrence_number times and that the average probability over of the triplet is greater than relation_probability. For details on the choice of relation_occurrence_number and relation_probability please refer to Appendix F. Finally, the category t with the highest probability is chosen as the final relation between e and d. Refer to Appendix C for more details.

4.7 Postprocessing

To map the model's output to a list of values for each medical entity, we initially filtered out the predictions with a probability score lower than threshold (denoted prediction_probability, more in Appendix F). Subsequently, to remove meaningless information, stop words and punctuation were excised from each predicted text.

Furthermore, our approach involved addressing instances where the model conveyed uncertainty or lacked adequate context. When the large language model (LLM) produced responses such as "I do not know" due to ambiguity or insufficiency, we systematically filtered out these outputs.

Further analysis revealed that models tend to generate the same answers in various forms depending on the given context. For instance, predictions such as "areds" and "areds-2 vitamins" essentially refer to the same value for a specific medical entity, but are expressed differently. To address these varia-

		Treatment		Factor		Coexists_with	
Architecture	Model	Precision	Recall	Precision	Recall	Precision	Recall
Encoder-only	RoBERTa-SQuAD-v2	0.25	0.54	0.21	0.75	0.3	0.14
	BioBERT-SQuAD-v2	0.13	0.9	0.25	0.75	0.45	0.71
	BERT-SQuAD-v2	0.17	0.45	0.17	0.45	0.17	0.57
Encoder-decoder	FLAN-T5-XXL: 0-shot	0.55	0.75	0.54	0.69	0.64	0.89
	FLAN-T5-XXL: few-shot	0.45	0.9	0.66	0.8	0.72	0.88
	FLAN-T5-XXL: instruct	0.86	0.9	0.8	0.8	0.83	0.97
	FLAN-T5-XXL: guided	0.88	**1**	0.82	**0.875**	0.76	0.875
	FLAN-UL2: 0-shot	0.43	0.9	0.16	0.62	0.74	0.85
	FLAN-UL2: few-shot	0.55	0.9	0.36	0.75	0.78	0.89
	FLAN-UL2: instruct	**0.98**	**1**	0.8	0.8	**0.98**	**1**
	FLAN-UL2: guided	**0.98**	**1**	**0.84**	**0.875**	**0.98**	**1**
Decoder-only	Vicuna-33B: guided	0.63	**1**	0.5	0.75	0.46	0.75
	Llama-2-70B: guided	0.65	**1**	0.38	0.75	0.4	0.875
	WizardLM-70B: guided	0.78	**1**	0.61	**0.875**	0.5	0.875

Table 2: We are comparing the performance of LLMs with various architectures across all three medical entities. The evaluation is based on precision and recall measurements for each medical entity within the final KG. The baseline for comparison are the entity values available in the notes. 'guided' refers to the guided instruction-based prompting described in Subsection 4.4.

tions, we employed normalized cosine similarity for the tokens in the model's predictions. Specifically, for each medical entity, we calculated the cosine similarity between each pair of predictions. Predictions which exceed the similarity threshold (denoted similarity_postprocessing, more in Appendix F) were considered equivalent and subsequently grouped together. From each group, the prediction with the highest initial probability score assigned by the model was selected. Finally, the refined output was converted into a list of values, selecting spans of text directly from the LLM output. A qualitative example illustrating this process is provided in Appendix D.

5 Results

We now describe our experimental study over our techniques for constructing the KG.

Setup We construct a KG for age-related macular degeneration (AMD), a progressive eye disease predominantly affecting older individuals with a high incidence rate. Since KGs are typically too large to display directly, we provide their tabular representation instead. To reconstruct KGs from Tables 3 and 4, connect treatments listed in the *Treatment* column to AMD (the target entity) using arrows. Similarly, connect factors from the *Factor* column to AMD, and establish connections to AMD with undirected edges for entities in the *Coexists_with* column.

The evaluation is based on precision and recall, which represent the ratio of correctly extracted terms by the model to all terms extracted by the model, and the ratio of correctly extracted terms by the model to all actual terms available in the clinical notes. The same metrics have been calculated for each entity (Treatment, Factor, and Coexists_with) separately. Therefore, the ground truth for comparison has been the entity values available in the clinical notes. Thus, we needed to review all clinical notes related to AMD and extract all factors, treatments, and 'coexists_with' terms. You can find the explanation of these terms in Subsection 4.5. The AMD-related notes have been identified according to Subsection 4.1 and preprocessed as described in Subsection 3.1. These steps leave us with 320 clinical notes.We refer to the process of extracting terms as annotation. This annotation was carried out by two of the authors, a retina specialist, and a clinical research coordinator. To establish a consistent annotation schema, a set of examples was jointly annotated. Following this, each annotator independently annotated the same set of examples, and the two sets of annotations were then combined via a joint manual adjudication process. As a result, we extracted 11 different treatments, 8 different factors, and 8 'Coexists_with' terms from the clinical notes.

Precision and recall results Table 2 shows the precision and recall results of different LLMs of various architectures. The best performance is consistently achieved with encoder-decoder LLMs for most medical entities. Specifically, FLAN-UL2, when used with our proposed prompt design, outperforms the other models. Furthermore, we observe that encoder-decoder models using 0-shot and few-shot prompting techniques are comparable to decoder-only models in some cases. However, when instruction-based or our proposed guided prompting is employed for encoder-decoder models, they significantly outperform the others.

Quantitative results for decoder-only models using 0-shot, few-shot, and instruction-based prompting techniques are not available. These models did not produce structured outputs, rendering them unsuitable for our task. Additional information can be found in Decoder-only models. Unlike other prompting techniques, guided instruction-based prompting (as described in Subsection 4.4) has demonstrated significant improvements. This prompt design allowed us to utilize only three decoder-only models for this task, out of the seven we experimented with. These models include Llama 2 (Touvron et al., 2023), Vicuna-33B (Zheng et al., 2023), and WizardLM-70B (Xu et al., 2023). The other four did not produce structured outputs with this prompt design, similar to the results obtained with the other three prompting techniques.

Notably, WizardLM-70B achieves the highest recall for factors and treatments, demonstrating that the incorporation of additional guidance has enhanced the understanding of the task by some of the decoder-only models, resulting in more precise and accurate answers. We believe that further research is required to explore the potential of decoder-only models for challenging relation extraction tasks, and future investigations may enhance their reliability. See prediction examples in Appendix E.

Decoder-only models Here we describe the challenges that make some of these models (BioGPT (Luo et al., 2022), OPT (Zhang et al., 2022), OPT-IML-MAX (Iyer et al., 2022), Bloom (Scao et al., 2022)) with any of the prompting techniques were unsuitable for clinical relation extraction, thus KG construction. Some of the models are prone to "hallucinating", a term commonly used to refer to the models generating responses that are factually incorrect or nonsensical. See such examples in Appendix E.2.1.

Furthermore, we observed cases where some models generated correct responses, but these responses did not originate from the given context. Another concern was the generation of excessively verbose or repetitive responses. Despite being contextually correct, the lengthy and redundant nature of these outputs complicated the postprocessing phase, making the integration of such responses into our KG construction pipeline impossible. See such examples in Appendix E.2.2.

Qualitative Example: AMD We continue using AMD as a qualitative example. AMD is a progressive eye disease affecting the retina, specifically the macula. The risk factors for AMD have been studied extensively and have widely been known to include age, race, smoking status, diet, and genetics (Holz et al., 2014; Heesterbeek et al., 2020). The exact reasons and mechanisms behind AMD are not yet fully researched. There are multiple pathways and factors for drusen formation and AMD progression, so it is hard to disentangle them. Large and numerous drusen are associated with an increased risk of developing advanced AMD (Schlanitz et al., 2019). The pathophysiologic landscape of AMD potentially involves degenerative transformations within several ocular components, including the outer retinal layers, the photoreceptors, retinal pigment epithelium (RPE) characterized by the loss of the ellipsoid zone (EZ) and atrophic changes, accumulation of subretinal/submacular fluids, perturbations in Bruch's membrane leading to choroidal neovascularization (CNVM), and areas of choriocapillaris nonperfusion resulting in macular atrophy and fibrosis (Holz et al., 2014; Boyer et al., 2017). Medical evaluators annotated drusen, genetics, CNVM, smoking, RPE irregularities, submacular/subretinal fluid, fibrosis, and loss of EZ zone as risk factors for AMD. The KG constructed with the utilization of FLAN-UL2 with guided instruction-based prompting that have relatively the best quantitative performance, is visually presented in Table 3.

Notably, besides factors, the graph also highlights a spectrum of terms that are linked to potential treatments and symptoms associated with AMD. Among the treatment entities are ARED-

Table 3: KG for AMD constructed using FLAN-UL2 model with guided instruction-based prompting. Red color indicates an incorrect values. Orange color indicates a values missed by the model.

Treatment	Factor	Coexists_with
AREDS vitamins	Drusen	Poor visual acuity
Avastin	Genetics / Family history	Metamorphopsia
Lucentis	Peripheral CNVM/CNVM	Visual changes
PDT	Smoking	Macula Risk genetic testing
WACS vitamins	RPE irregularity	Wet AMD
Amsler grid testing	Submacular fibrosis and fluid	Dry AMD/GA
Spinach	Loss of EZ zone	ForeseeHome
Fish	Glaucoma	Drusen
Omega-3 fatty acids	Subretinal fluid	Amblyopia
Anti-VEGF		
Green Leafy Vegetables		
Lack drusen		

S/WACS vitamins, dietary interventions, and Anti-VEGF treatments including Avastin and Lucentis. Other treatments indicated include PDT (Photodynamic Therapy), the utilization of Amsler grid, supplementation of Omega-3 fatty acids, and consumption of specific foods such as fish, spinach, and green leafy vegetables. The symptomatic aspects of AMD encompass a range of visual impairments and clinical manifestations. Patients afflicted with AMD often experience poor visual acuity, metamorphopsia (distorted vision), and can be diagnosed with either dry or wet AMD. Additionally, the management of the condition often involves undergoing assessments such as ForeseeHome and Macula Risk genetic testing, which play a pivotal role in monitoring the progression and development of AMD. Each of these terms is identified as values to the 'Coexists_with' entity within the graph.

Table 4: KG for AMD constructed using SemMedDB.

Treatment	Factor	Coexists_with
Injection procedure	Blind Vision	Visual impairment
Photochemotherapy	Antioxidants	Massive hemorrhage
Antioxidants	Oxidative Stress	Autofluorescence
Bevacizumab		Blindness
Eye care		Legal, Disability NOS
Homocysteine thiolactone		
Operative Surgical Procedures		

We also show the KG constructed by SemMedDB (Kilicoglu et al., 2012) in Table 4. SemMedDB is a repository of semantic predictions extracted from the titles and abstracts of all PubMed citations. It is evident that our approach has identified terms not found in the SemMedDB. Our method may not forge new terms where none existed in the original medical literature repository. However, the feedback from our medical evaluators underscores its potential to contribute to novel discoveries by highlighting existing but overlooked information.

6 Conclusion

In this paper, we propose an end-to-end approach that harnesses LLMs for the automatic generation of KGs from EMR notes. KGs hold significant value in numerous healthcare domains, including drug discovery and clinical trial design. The entities involved in the KG construction process encompass diseases, factors, treatments, and manifestations that co-occur with patients experiencing these diseases. Through extensive evaluation across various LLM architectures, we have demonstrated that encoder-decoder models outperform others in clinical relation extraction. Additionally, we emphasize the need for additional investigation into the suitability of decoder-only models for medical applications, particularly given their critical safety implications. Furthermore, we provide guided prompt design to utilize these models. We believe that an automated knowledge extraction method may deliver substantial benefits to the medical community and facilitate further research in the field.

7 Limitations

The findings of the research are subject to several limitations. The primary one is that our experiments were conducted on a single dataset focused on one specific disease. This limitation arises from the necessity of annotations by medical practitioners, a process that is highly time-consuming. Furthermore, due to the private nature of our dataset, we opted to use only open-source models to ensure data privacy and security. While this approach safeguards patient information, it may limit the performance benefits that could be gained from proprietary models. Additionally, we assumed only query access to large language models (i.e., no gradients). Fine-tuning LLMs on a relevant corpus could potentially enhance their performance and accuracy, and this remains an area for future exploration.

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. In *Proc. of EMNLP*, pages 1998–2022, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270.

David S Boyer, Ursula Schmidt-Erfurth, Menno van Lookeren Campagne, Erin C Henry, and Christopher Brittain. 2017. The pathophysiology of geographic atrophy secondary to age-related macular degeneration and the complement pathway as a therapeutic target. *Retina (Philadelphia, Pa.)*, 37(5):819.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Payal Chandak, Kexin Huang, and Marinka Zitnik. 2023. Building a knowledge graph to enable precision medicine. *Scientific Data*, 10(1):67.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *ArXiv preprint*, abs/2210.11416.

Limeng Cui, Haeseung Seo, Maryam Tabar, Fenglong Ma, Suhang Wang, and Dongwon Lee. 2020. DETERRENT: knowledge guided graph attention network for detecting healthcare misinformation. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 492–502. ACM.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL-HLT*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Martin Drancé, Marina Boudin, Fleur Mougin, and Gayo Diallo. 2021. Neuro-symbolic xai for computational drug repurposing. In *KEOD*, pages 220–225.

Sia Gholami and Mehdi Noori. 2021. Zero-shot open-book question answering. *ArXiv preprint*, abs/2111.11520.

Thomas J Heesterbeek, Laura Lorés-Motta, Carel B Hoyng, Yara TE Lechanteur, and Anneke I den Hollander. 2020. Risk factors for progression of age-related macular degeneration. *Ophthalmic and Physiological Optics*, 40(2):140–170.

Frank G Holz, Erich C Strauss, Steffen Schmitz-Valckenberg, and Menno van Lookeren Campagne. 2014. Geographic atrophy: clinical features and potential therapeutic approaches. *Ophthalmology*, 121(5):1079–1091.

Pere-Lluís Huguet Cabot and Roberto Navigli. 2021. REBEL: Relation extraction by end-to-end language generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2370–2381, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Srinivasan Iyer, Xi Victoria Lin, Ramakanth Pasunuru, Todor Mihaylov, Daniel Simig, Ping Yu, Kurt Shuster, Tianlu Wang, Qing Liu, Punit Singh Koura, et al. 2022. Opt-iml: Scaling language model instruction meta learning through the lens of generalization. *ArXiv preprint*, abs/2212.12017.

Bernal Jimenez Gutierrez, Nikolas McNeal, Clayton Washington, You Chen, Lang Li, Huan Sun, and Yu Su. 2022. Thinking about GPT-3 in-context learning for biomedical IE? think again. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4497–4512, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. In *Proc. of EMNLP*, pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

Halil Kilicoglu, Dongwook Shin, Marcelo Fiszman, Graciela Rosemblat, and Thomas C Rindflesch. 2012. Semmeddb: a pubmed-scale repository of biomedical semantic predications. *Bioinformatics*, 28(23):3158–3160.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv preprint*, abs/1907.11692.

Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. 2022. Biogpt: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6):bbac409.

Sungrim Moon, Serguei Pakhomov, Nathan Liu, James O Ryan, and Genevieve B Melton. 2014.

A sense inventory for clinical abbreviations and acronyms created using clinical notes and medical dictionary resources. *Journal of the American Medical Informatics Association*, 21(2):299–307.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2022. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874, Dublin, Ireland. Association for Computational Linguistics.

Galia Nordon, Gideon Koren, Varda Shalev, Benny Kimelfeld, Uri Shalit, and Kira Radinsky. 2019. Building causal graphs from medical literature and electronic medical records. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 1102–1109. AAAI Press.

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proc. of EMNLP*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Thomas C Rindflesch and Marcelo Fiszman. 2003. The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text. *Journal of biomedical informatics*, 36(6):462–477.

Maya Rotmensch, Yoni Halpern, Abdulhakim Tlimat, Steven Horng, and David Sontag. 2017. Learning a health knowledge graph from electronic medical records. *Scientific reports*, 7(1):5994.

Alberto Santos, Ana R Colaço, Annelaura B Nielsen, Lili Niu, Maximilian Strauss, Philipp E Geyer, Fabian Coscia, Nicolai J Wewer Albrechtsen, Filip Mundt, Lars Juhl Jensen, et al. 2022. A knowledge graph to interpret clinical proteomics data. *Nature biotechnology*, 40(5):692–702.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *ArXiv preprint*, abs/2211.05100.

Ferdinand Schlanitz, Bernhard Baumann, Stefan Sacu, Lukas Baumann, Michael Pircher, Christoph K Hitzenberger, and Ursula Margarethe Schmidt-Erfurth. 2019. Impact of drusen and drusenoid retinal pigment epithelium elevation size and structure on the integrity of the retinal pigment epithelium layer. *British Journal of Ophthalmology*, 103(2):227–232.

Andrea C Skelly, Joseph R Dettori, and Erika D Brodt. 2012. Assessing bias: the importance of considering confounding. *Evidence-based spine-care journal*, 3(01):9–12.

Anders Søgaard. 2021. *Explainable natural language processing*. Morgan & Claypool Publishers.

Mujeen Sung, Jinhyuk Lee, Sean Yi, Minji Jeon, Sungdong Kim, and Jaewoo Kang. 2021. Can language models be biomedical knowledge bases? In *Proc. of EMNLP*, pages 4723–4734, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Madhumita Sushil, Dana Ludwig, Atul J Butte, and Vivek A Rudrapatna. 2022. Developing a general-purpose clinical language inference model from a large corpus of clinical notes. *ArXiv preprint*, abs/2210.06566.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Stanford alpaca: An instruction-following llama model.

Yi Tay, Mostafa Dehghani, Vinh Q Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Dara Bahri, Tal Schuster, Steven Zheng, et al. 2022. Ul2: Unifying language learning paradigms. In *The Eleventh International Conference on Learning Representations*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *ArXiv preprint*, abs/2307.09288.

Milena Trajanoska, Riste Stojanov, and Dimitar Trajanov. 2023. Enhancing knowledge graph construction using large language models. *Preprint*, arXiv:2305.04676.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. In *Proc. of ICLR*. OpenReview.net.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions. *ArXiv preprint*, abs/2304.12244.

Seonghyeon Ye, Hyeonbin Hwang, Sohee Yang, Hyeongu Yun, Yireun Kim, and Minjoon Seo. 2023. In-context instruction learning. *ArXiv preprint*, abs/2302.14691.

Yitayal Yitayew. 2023. Flan-ul2: A new open source flan 20b with ul2. https://www.yitay.net/blog/flan-ul2-20b.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *ArXiv preprint*, abs/2205.01068.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *ArXiv preprint*, abs/2306.05685.

Appendix

A Models

Encoder-only models Our approach utilizes a fine-tuned question-answering model based on BERT (Devlin et al., 2019), specifically fine-tuned on the SQuAD v2 dataset (Rajpurkar et al., 2016). This model, which we refer to as BERT-SQuAD-v2, benefits from the core principles of BERT, including random token masking during pretraining to encourage contextual understanding.

Inspired by advancements in the BERT family, we also incorporate RoBERTa (Liu et al., 2019), which is improved upon Bert by introducing a new pretraining recipe that includes training for longer and on larger batches, randomly masking tokens at each epoch instead of just once during preprocessing, and removing the next-sentence prediction objective. We also consider BioBERT (Lee et al., 2020), which is a pre-trained BERT model which is trained on different combinations of general & biomedical domain corpora.

Decoder-only models BioGPT (Luo et al., 2022), a generative Transformer model tailored for biomedical literature, has shown remarkable results on several biomedical NLP benchmarks, including an impressive 78.2% accuracy on PubMedQA (Jin et al., 2019). However, our efforts to employ BioGPT for relation extraction were met with challenges. The model frequently hallucinated during inference, making it unsuitable for our specific application in relation extraction.

Open Pretrained Transformers (OPT) (Zhang et al., 2022) represents a comprehensive suite of decoder-only transformers designed for large-scale research. OPT-30B, a particular model from this suite, has been pre-trained predominantly on English text with some multilingual data from CommonCrawl. Sharing similarities with GPT-3, it uses a causal language modeling (CLM) objective. OPT-IML (Iyer et al., 2022) represents an advanced version of the OPT model, enhanced with Instruction Meta-Learning. It's been trained on an extensive collection known as the OPT-IML Bench, comprising roughly 2000 NLP tasks from 8 different benchmarks. Two variations exist: the standard OPT-IML trained on 1500 tasks, and OPT-IML-Max that covers all 2000 tasks.

BLOOM (Scao et al., 2022) stands as a sophisticated autoregressive Large Language Model (LLM), designed to produce coherent text across 46 languages and 13 programming languages, replicating human-like text generation capabilities.

Llama 2 (Touvron et al., 2023) is a distinguished collection of generative text models, with models ranging from 7 billion to 70 billion parameters. Presented by Meta, this repository encompasses the 70B variant, made compatible with the Hugging Face Transformers framework. Within the Llama 2 family lies a specialized series called Llama-2-70B-Chat, meticulously fine-tuned for dialogue-centric applications. This model excels, outstripping many open-source chat models in benchmarks and rivalling prominent closed-source counterparts like ChatGPT and PaLM in terms of helpfulness and safety.

Emerging from the wave of advanced chatbots, Vicuna-33B (Zheng et al., 2023) stands out as an open-source contribution, fine-tuned using the LLaMA framework based on dialogues from ShareGPT. Notably, when evaluated using GPT-4, Vicuna-33B not only showcased a commendable performance, rivaling the likes of OpenAI's ChatGPT and Google Bard (achieving over 90%* quality), but also surpassed counterparts like LLaMA and Stanford Alpaca (Taori et al., 2023) in over 90%* of the tests. This exceptional achievement comes at a modest training cost of around $300, making Vicuna-33B an attractive proposition. Additionally, its code, weights, and a live demo are accessible for the research community, albeit restricted to non-commercial applications.

WizardLM-70B (Xu et al., 2023) is a Large Language Model (LLM) built on the foundation of LLaMA, incorporating a novel training approach known as Evol-Instruct. This method involves leveraging artificial intelligence to evolve complex instruction data, setting WizardLM apart from LLaMA-based LLMs trained on simpler instructions. As a result it outperforms counterparts in tasks that demand intricate understanding and execution of instructions.

Encoder-decoder models FLAN-T5-XXL (Chung et al., 2022) is a encoder-decoder model that has been pre-trained on a multi-task mixture of unsupervised and supervised tasks and for which each task is converted into a text-to-text format. It performs well on multiple tasks including question answering.

FLAN-UL2 (Yitayew, 2023) is an encoder-decoder model based on the T5 architecture. It uses the same configuration as the UL2 (Tay et al., 2022) model released earlier last year and was fine-tuned using the "Flan" prompt tuning and dataset collection (Wei et al., 2022). According to the original blog, there are some notable improvements over the original UL2 model. The Flan-UL2 checkpoint uses a receptive field of 2048 which makes it more usable for few-shot in-context learning. This Flan-UL2 checkpoint does not require mode tokens anymore.

In comparison to FLAN-T5, FLAN-UL2 outperforms FLAN-T5 XXL on all four setups with an overall decent performance lift of +3.2% relative improvement. Most of the gains seem to come from the CoT setup while performance on direct prompting (MMLU and BBH) seems to be modest at best.

B Question list

Table 5: List of questions categorized by the medical entity. The "%s" in the questions represents a placeholder for a disease.

Medical entity	Question
Treatment	What can slow the progression of %s? (T1)
	What can decrease the chance of %s? (T2)
	What can reduce the risk of %s? (T3)
	What is a treatment for %s? (T4)
	What treats %s? (T5)
Factor	What does cause %s? (F1)
	What is the cause of %s? (F2)
	What is the factor for %s? (F3)
	What can increase the risk of %s? (F4)
	What can convert to %s? (F5)
Effect	What can %s convert to? (E1)
	What is the effect of %s? (E2)
	What does %s lead to? (E3)
	What can %s become? (E4)
	What does %s affect? (E5)

C Algorithms

Algorithm 1 Disease-specific notes identification

```
Ensure: $result$
  $result := \{\}$
  $D_{input}$ = UMLS_Metathesaurus_API($d_{input}$)
  for $note$ in $clinical\_notes$ do
      $D_{note}$ = BioBERT_NER($note$)
      for $d_i \in D_{input}$ do
          for $d_{note_i} \in D_{note}$ do
              if $note$ contains $d_i$ then
                  $result$.append($note$)
              else
                  $similarity\_score$ := calculate_cosine_similarity($d_{input}$, $d_{note_i}$)
              end if
              if $similarity\_score > threshold$ then
                  $result$.append($note$)
              end if
          end for
      end for
  end for
```

Algorithm 2 Querying LLM

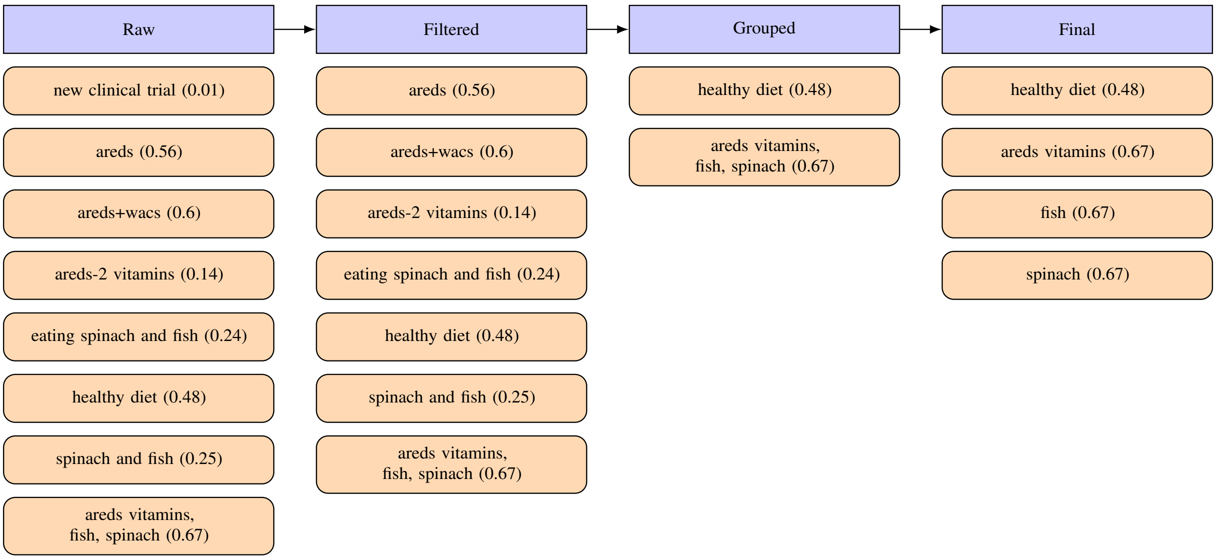
```
Ensure: $result$
  $result := \{\}$
  for $d \in D$ do
      for $c \in C(d)$ do
          for $t \in \{treatment, factor, coexists\_with\}$ do
              for $q(d) \in Q_t(d)$ do
                  $tmp := \langle LM(\langle c, q(d) \rangle), d, t \rangle$        ▷ Where LM returns a list of possible answers
                                                                   ▷ with their probabilities.
                  $result.insert(tmp)$
              end for
          end for
      end for
  end for
```

Algorithm 3 Relation extraction

```
Require: $result$ from Algorithm 2
  $relation := \{\}$
Ensure: relations
  for unique $\langle e, d, t \rangle$ in $result$ do
      $temp := \langle average(result[e, d, t].score), count(result[e, d, t]) \rangle$
      if $temp.average \geq 0.1$ and $temp.count \geq 10$ then
          $stat \leftarrow \langle temp.average, e, d, t \rangle$
      end if
  end for
  for unique $e, d$ in $stat$ do
      $relations \leftarrow \langle d, \arg\max_t stat[e, d], e \rangle$
  end for
```

D Postprocessing

Figure 2: Qualitative example of the postprocessing steps. Every orange node illustrates the predictions made by an LLM, along with an associated probability enclosed in parentheses.



E Prompts and sample outputs

E.1 Encoder-only models

E.1.1 Examples of wrong predictions

Listing 1: BERT-SQuAD-v2: wrong prediction

```
Question: What can slow the progression of
    macular degeneration?
Context: Macular Degeneration: Discussed the
    nature of dry macular degeneration.
    Discussed Age Related Eye Disease Study and
    recommended AREDs vitamins for prevention
    purposes. Patient given Amsler grid to
    monitor for metamorphopsias or changes in
    central vision.
Answer:
dry macular degeneration
```

Listing 2: RoBERTa-SQuAD-v2: wrong prediction

```
Question: What does cause Macular Degeneration?
Context: Macular Degeneration: Discussed the
    nature of dry macular degeneration.
    Discussed Age Related Eye Disease Study and
    recommended AREDs vitamins for prevention
    purposes. Patient given Amsler grid to
    monitor for metamorphopsias or changes in
    central vision.
Answer:
dry macular degeneration
```

E.1.2 Examples of correct predictions

Listing 3: BERT-SQuAD-v2: correct prediction

```
Question: What can slow the progression of
    myopic macular degeneration?
Context: Myopic Macular Degeneration - Explained
     that there is no specific treatment at this
     time. AREDS and WACS vitamins MAY help slow
     down the progression of the degeneration.
    Monitor closely. All questions were answered
     to the patient's satisfaction.
Answer:
AREDS and WACS vitamins
```

Listing 4: BioBERT-SQuAD-v2: correct prediction

```
Question: What does cause ARMD?
Context: Macular Degeneration: Discussed the
    nature of dry macular degeneration. Patient
    encouraged to use an Amsler grid to monitor
    macular function by looking for
    metamorphopsias or visual changes.
Answer:
metamorphopsias or visual changes
```

Listing 5: RoBERTa-SQuAD-v2: correct prediction

```
Question: What does cause amd?
Context: Smoking is the largest `modifiable'
    risk factor for AMD - by stopping smoking a
    patient can reduce the chance of getting AMD
     and of AMD progressing.There is a very
    strong association between smoking and
    macular disease. Smokers are up to four
    times more likely than non-smokers to
    develop age-related macular degeneration (
    AMD) and smokers with a genetic
    predisposition to AMD are eight times more
    likely to get the condition. Some research
    suggests people with certain genes can be 20
     times more likely to get macular
    degeneration if they smoke.
Answer:
Smoking
```

E.2 Decoder-only models

E.2.1 Examples of hallucinations

Listing 6: BLOOM with instruction-based prompting: hallucination

```
I want you to act as a question answering
    machine. I will provide you with a questions
     and a context and you will reply with the
    answers.
Question: What can slow the progression of armd?
Instruction: If the answer is not in context
    answer "i do not know".
Context: Wet ARMD- Patient educated on condition.
     Discussed diet and use of Macular
    Degeneration vitamins. All questions were
    answered to the patient's satisfaction.
Answer:
i do not know

    a:

    i think the question is a bit vague. it is
    not clear what the question is about. it is
    not clear what the context is. it is not
    clear what the answers are. it is not clear
    what the question is asking. it is not clear
    what the question is asking the answerer to
    do. it is not clear what the question is
    asking the answerer to do. it is not clear
    what the question is asking the answerer to
    do. it is not clear what the
```

Listing 7: BLOOM with guided instruction-based prompting: hallucination

```

You are a helpful medical knowledge extractor
    assistant. Your task is to answer to the
    question, using only and only the provided
    context. Do not use any other sources. Do
    not use your knowledge. Answer to the
    question using the context. Do not write
    explanation. Do not write long sentences.
    Answer with extracted entities, separated
    with commas. I will provide the inputs in
    the following format

### Input
question_type: treat/coexists_with/factor
question: [QUESTION]
context: [CONTEXT]

I want you to answer to the question in the
    following format, depending on the [
    ENTITY_TYPE] which can be either [TREAT], [
    COEXISTS_WITH], [FACTOR]. Do not include
    other entity types. Do not include treatment
     if not asked. Only extract entities for
    that type.
```

```
### Response

[ENTITY_TYPE]: [ENTITY_1], [ENTITY_2], [ENTITY_3
    ]...

If the question is not related to the context,
    or if the context contains a missing
    information to answer to the question,
    answer "I do not know." and nothing more. No
     explanations. If the context does not
    contain the answer to the question, answer
    with the following format

### Response
I do not know.

### Input

question_type: coexists_with
question: What does armd lead to?
context: Acute Exudative ARMD OU - chronic
    leakage OU. No significant changes OU.
    Recommend treating with Anti-VEGF injections
    . Discussed with patient the gravity of this
     disease, including the potential for vision
     loss and guarded recovery. Reviewed
    treatment options - Avastin, Lucentis, Eylea
    , PDT.

### Response
effect:

    If the question contains multiple entities,
    extract all of them, separated with commas.
    For example, if you have following question

    Question 1:
```

Listing 8: BioGPT with guided instruction-based prompting: hallucination

```
You are a helpful medical knowledge extractor
    assistant. Your task is to answer to the
    question, using only and only the provided
    context. Do not use any other sources. Do
    not use your knowledge. Answer to the
    question using the context. Do not write
    explanation. Do not write long sentences.
    Answer with extracted entities, separated
    with commas. I will provide the inputs in
    the following format

### Input
question_type: treat/coexists_with/factor
question: [QUESTION]
context: [CONTEXT]

I want you to answer to the question in the
    following format, depending on the [
    ENTITY_TYPE] which can be either [TREAT], [
    COEXISTS_WITH], [FACTOR]. Do not include
    other entity types. Do not include treatment
     if not asked. Only extract entities for
    that type.

### Response

[ENTITY_TYPE]: [ENTITY_1], [ENTITY_2], [ENTITY_3
    ]...

If the question is not related to the context,
    or if the context contains a missing
    information to answer to the question,
    answer "I do not know." and nothing more. No
     explanations. If the context does not
    contain the answer to the question, answer
    with the following format

### Response
I do not know.

### Input

question_type: factor
question: What is the factor for armd?
context: 2 small Druse OD- clinically does not
    look like ARMD. Patient has a family history
     of ARMD, recommend starting on AREDS + WACS
     vitamins. Eat green leafy vegetables like
    Spinach 5 times a week and fish at least 2
    times a week.

### Response
I do not include any other entities to answer the
    question.
```

Listing 9: OPT-30B with guided instruction-based prompting: hallucination

```
You are a helpful medical knowledge extractor
    assistant. Your task is to answer to the
    question, using only and only the provided
    context. Do not use any other sources. Do
    not use your knowledge. Answer to the
    question using the context. Do not write
    explanation. Do not write long sentences.
    Answer with extracted entities, separated
    with commas. I will provide the inputs in
    the following format

### Input
question_type: treat/coexists_with/factor
question: [QUESTION]
context: [CONTEXT]

I want you to answer to the question in the
    following format, depending on the [
    ENTITY_TYPE] which can be either [TREAT], [
    COEXISTS_WITH], [FACTOR]. Do not include
    other entity types. Do not include treatment
     if not asked. Only extract entities for
    that type.

### Response

[ENTITY_TYPE]: [ENTITY_1], [ENTITY_2], [ENTITY_3
    ]...

If the question is not related to the context,
    or if the context contains a missing
    information to answer to the question,
    answer "I do not know." and nothing more. No
     explanations. If the context does not
    contain the answer to the question, answer
    with the following format

### Response
I do not know.
```

```
### Input

question_type: coexists_with
question: What can amd convert to?
context: Explained to patient that he does have
     AMD but it is mild and not the cause of his
     blurred vision.

### Response
effect: [ENTITY_1], [ENTITY_2], [ENTITY_3]...

     If the question is not related to the
     context, or if the context contains a
     missing information to answer to the
     question, answer Ï do not know.änd nothing
     more. No explanations. If the context does
     not contain the answer to the question,
     answer with the following format

     ### Response

     I do not know.

     ### Input

     question_type: factor

     question: What can amd convert to?

     context: Explained to patient that he does
     have AMD but it is mild and not the cause of
     his blurred vision.

     ### Response

     factor: [ENTITY_1], [ENTITY_2],
     [ENTITY_3]...

     If the question is not related to the
     context, or if the context contains a
     missing information to answer to the
     question, answer Ï do not know.änd nothing
     more. No explanations. If the context does
     not contain the answer to the question,
     answer with the following format

     ### Response

     I do not know.

     ### Input

     question_type: treat

     question: What can amd convert to?

     context: Explained to patient that he does
     have AMD but it is mild and not the cause of
     his blurred vision.

     ### Response

     treat: [ENTITY_1], [ENTITY_2], [ENTITY_3]...

     If the question is not related to the
     context, or if the context contains a missing
     information to answer to the question,
     answer Ï do not know.änd nothing more.

     No explanations.

     If the context does not contain the answer
     to the question, answer with the following
     format

     ### Response

     I do not know.

     ### Input

     question_type: factor

     question: What can amd convert to?

     context: Explained to patient that he does
     have AMD but it is mild and
```

E.2.2 Examples of wrong predictions

Listing 10: OPT-IML-MAX-30B with guided instruction-based prompting: wrong prediction

```
You are a helpful medical knowledge extractor
     assistant. Your task is to answer to the
     question, using only and only the provided
     context. Do not use any other sources. Do
     not use your knowledge. Answer to the
     question using the context. Do not write
     explanation. Do not write long sentences.
     Answer with extracted entities, separated
     with commas. I will provide the inputs in
     the following format

### Input
question_type: treat/coexists_with/factor
question: [QUESTION]
context: [CONTEXT]

I want you to answer to the question in the
     following format, depending on the [
     ENTITY_TYPE] which can be either [TREAT], [
     COEXISTS_WITH], [FACTOR]. Do not include
     other entity types. Do not include treatment
      if not asked. Only extract entities for
     that type.

### Response

[ENTITY_TYPE]: [ENTITY_1], [ENTITY_2], [ENTITY_3
     ]...

If the question is not related to the context,
     or if the context contains a missing
     information to answer to the question,
     answer "I do not know." and nothing more. No
      explanations. If the context does not
     contain the answer to the question, answer
     with the following format

### Response
I do not know.

### Input

question_type: coexists_with
question: What can armd convert to?
context: Dry ARMD - Explained that there is no
     specific treatment at this time. Can convert
      to wet ARMD, which is treatable. Eating a
     healthy diet that includes green leafy
     vegetables, fish, taking AREDS + WACS eye
     vitamins and Omega-3 fatty acids has been
     shown to significantly decrease the
     progression of the disease. Monitor closely,
      daily use of amsler grid. All questions
     were answered to the patient's satisfaction

### Response
I do not know
```

Listing 11: OPT-IML-MAX-30B with guided instruction-based prompting: wrong prediction

```
You are a helpful medical knowledge extractor
     assistant. Your task is to answer to the
     question, using only and only the provided
     context. Do not use any other sources. Do
     not use your knowledge. Answer to the
     question using the context. Do not write
     explanation. Do not write long sentences.
     Answer with extracted entities, separated
```

```
    with commas. I will provide the inputs in
    the following format

### Input
question_type: treat/coexists_with/factor
question: [QUESTION]
context: [CONTEXT]

I want you to answer to the question in the
    following format, depending on the [
    ENTITY_TYPE] which can be either [TREAT], [
    COEXISTS_WITH], [FACTOR]. Do not include
    other entity types. Do not include treatment
     if not asked. Only extract entities for
    that type.

### Response

[ENTITY_TYPE]: [ENTITY_1], [ENTITY_2], [ENTITY_3
    ]...

If the question is not related to the context,
    or if the context contains a missing
    information to answer to the question,
    answer "I do not know." and nothing more. No
     explanations. If the context does not
    contain the answer to the question, answer
    with the following format

### Response
I do not know.

### Input

question_type: coexists_with
question: What does armd lead to?
context: Acute Exudative ARMD OU - chronic
    leakage OU. No significant changes OU.
    Recommend treating with Anti-VEGF injections
    . Discussed with patient the gravity of this
     disease, including the potential for vision
     loss and guarded recovery. Reviewed
    treatment options - Avastin, Lucentis, Eylea
    , PDT.

### Response
I do not know
```

Listing 12: Vicuna-33B with instruction-based prompting: wrong prediction

```

Instruction: I want you to act as a medical
    question answering machine. I will provide
    you with
questions and a context, and you will reply with
     the answers. If the answer is not in
    context, answer "I do not know."
question: What can decrease the chance of armd?
context: Acute Exudative ARMD OS - Subertinal
    blood/exudates OU-Discussed with patient the
     gravity of this disease, including the
    potential for vision loss and guarded
    recovery. Reviewed treatment options -
    Avastin, Lucentis, Eylea, PDT.

Answer:
In the context provided, the question is about
    decreasing the chance of ARMD (Age-Related
    Macular Degeneration). While I am not a
    doctor, I can provide some general
    information on this topic.There are certain
    risk factors for ARMD that you may not be
    able to change, such as age, family history,
    and genetics. However, there are some
    lifestyle changes that may help decrease the
    risk of ARMD or slow its
    progression:Maintain a healthy diet rich in
    fruits, vegetables, and whole grains. Foods
    high in antioxidants, such as vitamins C and
    E, zinc, and lutein, may be particularly
    beneficial.Exercise regularly, as studies
    have shown that physical activity may reduce
    the risk of ARMD.Maintain a healthy weight,
    as obesity has been linked to a higher risk
    of ARMD.Quit smoking, as smoking is a risk
    factor for ARMD and can exacerbate the
    condition.Regularly check your vision and
    have annual eye exams, especially if you
    have a family history of ARMD or other risk
    factors. Early detection and treatment can
    help manage the condition more
    effectively.It is important to discuss these
    options with a healthcare professional to
    determine the best course of action for your
    specific situation.
```

Listing 13: LLama-2-70B-chat with instruction-based prompting: wrong prediction

```
Instruction: I want you to act as a medical
    question answering machine. I will provide
    you with
questions and a context, and you will reply with
     the answers. If the answer is not in
    context, answer "I do not know."
question: What can heal amd? # TODO change heal
    to our question
context: Wet ARMD (Inactive) OD - Improved, no
    active leakage was seen during dilated eye
    exam and diagnostic testing. Patient is
    advised to continue AREDS + WACS vitamins.
    Eat green leafy vegetables like Spinach 5
    times a week and fish at least 2 times a
    week.Recommend close monitoring and
    treatment as indicated. Patient will return
    to office in 1 month for IVFA to monitor AMD
    .
Answer:
I cannot provide medical advice or recommend
    treatments for specific individuals. However,
    I can provide some general information about
    age-related macular degeneration (AMD) and
    its treatment options.AMD is a chronic eye
    condition that affects the macula, the part
    of the retina responsible for central vision.
    There are two main types of AMD: dry and wet.
    Wet AMD is characterized by the growth of
    new blood vessels under the retina, which
    can leak fluid and cause vision loss.While
    there is no cure for AMD, there are several
    treatment options available to slow down its
    progression and manage its symptoms. For wet
    AMD, anti-vascular endothelial growth factor
    (VEGF) injections are commonly used to
    prevent further vision loss. These
    medications work by blocking the growth of
    new blood vessels and reducing the amount of
    fluid in the retina.In addition to
    medication, a healthy lifestyle can also
    help manage AMD. A diet rich in fruits,
    vegetables, and omega-3 fatty acids, along
    with regular exercise and not smoking, can
    reduce the risk of AMD progression.It's
    important to note that the best course of
    treatment for AMD will depend on the
    individual patient and the severity of their
    condition. Regular eye exams and
    consultations with an eye care professional
    are necessary to determine the most
    appropriate treatment plan.In the context of
    the patient you described, the recommended
    treatment plan includes continuing AREDS +
    WACS vitamins, eating a healthy diet, and
```

```
     close monitoring with regular eye exams.
     IVFA (intravitreal fusion) may also be
     recommended to monitor the progression of
     AMD and determine if additional treatment is
     needed. However, it's important to consult
     with an eye care professional for
     personalized medical advice.
```

Listing 14: LLama-2-70B-chat with instruction-based prompting: wrong prediction

```
Instruction: I want you to act as a medical
     question answering machine. I will provide
     you with
questions and a context, and you will reply with
      the answers. If the answer is not in
     context, answer "I do not know."
question: What can reduce the risk of armd?
context: Acute Exudative ARMD OS - ICG/IVFA
     shows active bleeding. Discussed with
     patient the gravity of this disease,
     including the potential for vision loss and
     guarded recovery. Reviewed treatment options
      - Avastin, Lucentis, Eylea, PDT.
Answer:
I do not know.The question asked is not in
     context with the given situation. The
     patient has acute exudative age-related
     macular degeneration (ARMD) with active
     bleeding, and the doctor has already
     discussed the severity of the disease and
     the potential treatment options with the
     patient. The question about what can reduce
     the risk of ARMD is not relevant to the
     current situation.
```

E.2.3 Examples of correct predictions

Listing 15: WizardLM-70B with guided instruction-based prompting: correct prediction

```
You are a helpful medical knowledge extractor
     assistant. Your task is to answer to the
     question, using only and only the provided
     context. Do not use any other sources. Do
     not use your knowledge. Answer to the
     question using the context. Do not write
     explanation. Do not write long sentences.
     Answer with extracted entities, separated
     with commas. I will provide the inputs in
     the following format

### Input
question_type: treat/coexists_with/factor
question: [QUESTION]
context: [CONTEXT]

I want you to answer to the question in the
     following format, depending on the [
     ENTITY_TYPE] which can be either [TREAT], [
     COEXISTS_WITH], [FACTOR]. Do not include
     other entity types. Do not include treatment
      if not asked. Only extract entities for
     that type.

### Response

[ENTITY_TYPE]: [ENTITY_1], [ENTITY_2], [ENTITY_3
     ]...

If the question is not related to the context,
     or if the context contains a missing
     information to answer to the question,
     answer "I do not know." and nothing more. No
      explanations. If the context does not
     contain the answer to the question, answer
     with the following format

### Response
I do not know.

### Input

question_type: factor
question: What can increase the risk of armd?
context: 2 small Druse OD- clinically does not
     look like ARMD. Patient has a family history
      of ARMD, recommend starting on AREDS + WACS
      vitamins. Eat green leafy vegetables like
     Spinach 5 times a week and fish at least 2
     times a week.

### Response
factor: factor: family history, diet, lifestyle
```

Listing 16: Vicuna-33B with guided instruction-based prompting: correct prediction

```
You are a helpful medical knowledge extractor
     assistant. Your task is to answer to the
     question, using only and only the provided
     context. Do not use any other sources. Do
     not use your knowledge. Answer to the
     question using the context. Do not write
     explanation. Do not write long sentences.
     Answer with extracted entities, separated
     with commas. I will provide the inputs in
     the following format

### Input
question_type: treat/coexists_with/factor
question: [QUESTION]
context: [CONTEXT]

I want you to answer to the question in the
     following format, depending on the [
     ENTITY_TYPE] which can be either [TREAT], [
     COEXISTS_WITH], [FACTOR]. Do not include
     other entity types. Do not include treatment
      if not asked. Only extract entities for
     that type.

### Response

[ENTITY_TYPE]: [ENTITY_1], [ENTITY_2], [ENTITY_3
     ]...

If the question is not related to the context,
     or if the context contains a missing
     information to answer to the question,
     answer "I do not know." and nothing more. No
      explanations. If the context does not
     contain the answer to the question, answer
     with the following format

### Response
I do not know.

### Input
```

```
question_type: treat
question: What can decrease the chance of armd?
context: Acute Exudative ARMD OS - Subertinal blood/exudates OU-Discussed with patient the gravity of this disease, including the potential for vision loss and guarded recovery. Reviewed treatment options - Avastin, Lucentis, Eylea, PDT.

### Response
treat: Avastin, Lucentis, Eylea, PDT
```

Listing 17: LLama-2-70B-chat with guided instruction-based prompting: correct prediction

```
You are a helpful medical knowledge extractor assistant. Your task is to answer to the question, using only and only the provided context. Do not use any other sources. Do not use your knowledge. Answer to the question using the context. Do not write explanation. Do not write long sentences. Answer with extracted entities, separated with commas. I will provide the inputs in the following format

### Input
question_type: treat/coexists_with/factor
question: [QUESTION]
context: [CONTEXT]

I want you to answer to the question in the following format, depending on the [ENTITY_TYPE] which can be either [TREAT], [COEXISTS_WITH], [FACTOR]. Do not include other entity types. Do not include treatment if not asked. Only extract entities for that type.

### Response

[ENTITY_TYPE]: [ENTITY_1], [ENTITY_2], [ENTITY_3]...

If the question is not related to the context, or if the context contains a missing information to answer to the question, answer "I do not know." and nothing more. No explanations. If the context does not contain the answer to the question, answer with the following format

### Response
I do not know.

### Input

question_type: treat
question: What can slow the progression of armd?
context: Dry ARMD OU- Explained that there is no specific treatment at this time. Can convert to wet ARMD, which is treatable. Eating a healthy diet that includes green leafy vegetables, fish, taking AREDS + WACS eye vitamins and Omega-3 fatty acids has been shown to significantly decrease the progression of the disease. Monitor closely. All questions were answered to the patient's satisfaction.

### Response
treat: AREDS + WACS eye vitamins, Omega-3 fatty acids, healthy diet including green leafy vegetables, fish
```

E.3 Encoder-decoder models

E.3.1 Examples of wrong predictions

Listing 18: FLAN-UL2 with instruction-based few-shot prompting: wrong prediction

```
Instruction: I want you to act as a question answering machine. I will provide you with a question and a context, and you will reply with the answers.
Question: What can slow the progression of AMD?
Context: Macular Dystrophy vs. Early Dry AMD OU - Explained that there is no specific treatment at this time. Patient educated on condition. Eating a healthy diet that includes green leafy vegetables, fish, taking AREDS + WACS eye vitamins and Omega-3 fatty acids has been shown to significantly decrease the progression of the disease.
Answer: Eating a healthy diet that includes green leafy vegetables.

Question: What can slow the progression of myopic macular degeneration?
Context: Myopic Macular Degeneration - Explained that there is no specific treatment at this time. AREDS and WACS vitamins MAY help slow down the progression of the degeneration. Monitor closely. All questions were answered to the patient's satisfaction.
Answer: AREDS and WACS vitamins

Question: What can myopic macular degeneration convert to?
Context: Myopic Macular Degeneration - Explained that there is no specific treatment at this time. AREDS and WACS vitamins MAY help slow down the progression of the degeneration. Monitor closely. All questions were answered to the patient's satisfaction.
Answer: AREDS + WACS eye vitamins
```

E.3.2 Examples of correct predictions

Listing 19: FLAN-T5-XXL with instruction-based prompting: correct prediction

```
I want you to act as a question answering machine. I will provide you with a questions and a context and you will reply with the answers.
Question: What can slow the progression of armd?
Instruction: If the answer is not in context answer "i do not know".
Context: Wet ARMD- Patient educated on condition. Discussed diet and use of Macular Degeneration vitamins. All questions were answered to the patient's satisfaction.
Answer:
vitamins
```

Listing 20: FLAN-T5-XXL with few-shot prompting: correct prediction

```
question: What can slow the progression of
    macular disease?
context: very strong association between smoking
     and macular disease. Smokers are up to four
     times more likely than non-smokers to
    develop age-related macular degeneration (
    AMD) and smokers with a genetic
    predisposition to AMD are eight times more
    likely to get the condition. Some research
    suggests people with certain genes can be 20
     times more likely to get macular
    degeneration if they smoke.
target: the answer to the question given the
    context is smoking.

question: What can slow the progression of amd?
context: Macular Dystrophy vs. Early Dry AMD OU -
     Explained that there is no specific
    treatment at this time. Patient educated on
    condition. Eating a healthy diet that
    includes green leafy vegetables, fish,
    taking AREDS + WACS eye vitamins and Omega-3
     fatty acids has been shown to significantly
     decrease the progression of the disease.
    Stressed the need for follow up exams. All
    questions were answered to the patient's
    satisfaction.
target: the answer to the question given the
    context is Eating a healthy diet that
    includes green leafy vegetables.

question: What can slow the progression of
    myopic macular degeneration?
context: D/w pt: Myopic macular degeneration.
    Diagnosis discussed with patient. Possible
    treatments explained including glasses,
    refractive surgery, contact lenses or doing
    nothing. All questions were answered to
    patients satisfaction.
target: the answer to the question given the
    context is glasses
```

Listing 21: FLAN-UL2 with few-shot prompting: correct prediction

```
very strong association between smoking and
    macular disease. Smokers are up to four
    times more likely than non-smokers to
    develop age-related macular degeneration (
    AMD) and smokers with a genetic
    predisposition to AMD are eight times more
    likely to get the condition. Some research
    suggests people with certain genes can be 20
     times more likely to get macular
    degeneration if they smoke.
Create a bulleted list of what can slow the
    progression of macular disease?
- not smoking

Macular Dystrophy vs. Early Dry AMD OU -
    Explained that there is no specific
    treatment at this time. Patient educated on
    condition. Eating a healthy diet that
    includes green leafy vegetables, fish,
    taking AREDS + WACS eye vitamins and Omega-3
     fatty acids has been shown to significantly
     decrease the progression of the disease.
    Stressed the need for follow up exams. All
    questions were answered to the patient's
    satisfaction. target: the answer to the
    question given the context is Eating a
    healthy diet that includes green leafy
    vegetables.
Create a bulleted list of what can slow the
    progression of amd?
- Eating a healthy diet
- Green leafy vegetables

Myopic Macular Degeneration - Explained that
    there is no specific treatment at this time.
     AREDS and WACS vitamins MAY help slow down
    the progression of the degeneration. Monitor
     closely. All questions were answered to the
     patient's satisfacti
Create a bulleted list of What can slow the
    progression of myopic macular degeneration?.
- AREDS
- WACS vitamins
```

Listing 22: FLAN-UL2 with instruction prompting: correct prediction

```
Instruction: I want you to act as a medical
    question answering machine. I will provide
    you with
questions and a context, and you will reply with
     the answers.
Question: What does armd affect?
Instruction: If the answer is not in context,
    answer "I do not know."
Context: Acute Exudative ARMD/ CSCR OD - appears
     slightly worse on OCT and exam. Reviewed
    treatment options - Avastin, Lucentis, Eylea
    , PDT.
Answer: I do not know
```

F Implementation Details

Hyperparameter	Suggested Value	Intuition
threshold_preprocessing	0.8	Aims to accurately identify and include only those clinical notes that are directly relevant to the diseases being studied. Higher thresholds excluded valuable information, so we ensured a comprehensive dataset without compromising on relevance.
threshold_notes_identification	0.8	A disease may have multiple textual representations in general and it may be written in different ways by different clinicians (some terms may be abbreviated, some may contain typos). This threshold is used to understand if a disease entry written by a clinician matched with a set of standard forms of the disease by computing their cosine similarity and if the value is above this threshold that it is considered to be the same disease and the note to be containing relevant information about the disease.
similarity_postprocessing	0.8	Applied to address variations in how models express the same medical entities, such as "areds" versus "areds-2 vitamins". By calculating the normalized cosine similarity between each pair of predictions and grouping those with a similarity score exceeding 0.8, we effectively identify and consolidate equivalent predictions. This threshold not only enhances the consistency but also maintains its comprehensiveness by filtering out responses that indicate uncertainty or lack sufficient context.
relation_occurrence_number	10	Balances between reliability and inclusivity. This threshold ensures that the relation is not an outlier or a random occurrence, contributing to the robustness of the KG. It is chosen to filter out infrequent relations that might be anomalies or errors, while still allowing less common but valid relations to be included.
relation_probability	0.1	Ensures to capture a wide array of potential relationships within the biomedical context. This inclusivity is essential for identifying both prominent and subtle relations that may not be immediately apparent in the data but are nevertheless significant.
prediction_probability	0.08	Balances the removal of low-confidence predictions, which might represent noise or uncertain information, while retaining those with a reasonable likelihood of accuracy.

Table 6: Hyperparameters of the system.

All the method's hyperparameters have been selected through experimentation with the data and may be adjusted for the specific dataset being utilized. Further explanation on the rationale behind the selection of each hyperparameter is provided in Table 6.

Document-level Clinical Entity and Relation Extraction via Knowledge Base-Guided Generation

Kriti Bhattarai[1,2], Inez Y. Oh[1], Zachary B. Abrams[1], Albert M. Lai[1,2]
Institute for Informatics, Data Science & Biostatistics, Washington University School of Medicine
Department of Computer Science, Washington University in St. Louis

Abstract

Generative pre-trained transformer (GPT) models have shown promise in clinical entity and relation extraction tasks because of their precise extraction and contextual understanding capability. In this work, we further leverage the Unified Medical Language System (UMLS) knowledge base to accurately identify medical concepts and improve clinical entity and relation extraction at the document level. Our framework selects UMLS concepts relevant to the text and combines them with prompts to guide language models in extracting entities. Our experiments demonstrate that this initial concept mapping and the inclusion of these mapped concepts in the prompts improves extraction results compared to few-shot extraction tasks on generic language models that do not leverage UMLS. Further, our results show that this approach is more effective than the standard Retrieval Augmented Generation (RAG) technique, where retrieved data is compared with prompt embeddings to generate results. Overall, we find that integrating UMLS concepts with GPT models significantly improves entity and relation identification, outperforming the baseline and RAG models. By combining the precise concept mapping capability of knowledge-based approaches like UMLS with the contextual understanding capability of GPT, our method highlights the potential of these approaches in specialized domains like healthcare.

1 Introduction

Generative pre-trained transformer (GPT) models have shown significant potential across various clinical tasks, including information extraction, summarization, and question-answering (Agrawal et al., 2022; Tang et al., 2023a; Yang et al., 2022, Singhal et al., 2023). Generative models are able to generate contextually relevant text given a prompt. However, for real-world clinical use, in tasks that require high precision, it is equally important to understand the context and minimize the errors that come from GPT models. However, accuracy of these models is limited to their training data. While GPT models are great at capturing nuanced contextual information, they often fall short in accurately identifying all medical concepts, possibly due to limited or outdated domain-specific data (Tang et al., 2023b, Singhal et al., 2023).

Knowledge bases store domain-specific data. Medical knowledge bases, such as, Unified Medical Language System (UMLS) knowledge base (Bodenreider, 2004), include comprehensive information about medical concepts. Integrating knowledge bases with language models is an open research area with multiple works exploring different ways of integrating them with language models, such as BERT (Devlin et al., 2019). There are limited studies on the integration of medical knowledge bases, particularly UMLS, with most recent large language models (LLMs), such as GPT.

To address this limitation, we introduce an approach for clinical entity extraction that leverages UMLS for knowledge augmentation. While GPT can identify nuanced contextual information, UMLS includes a comprehensive repository of domain-specific clinical concepts that GPT may not recognize, such as brand names for drugs, abbreviations, acronyms, and aliases (Agrawal et al., 2022).

Our contributions in this paper are summarized as follows:
(1) we introduce a framework to integrate UMLS concepts into the default generative models to facilitate few-shot information extraction of biomedical entities and relations.
(2) we explore current state-of-the-art knowledge augmentation techniques, such as Retrieval Augmented Generation (RAG) aimed at improving extraction, and
(3) we conduct evaluation of our framework, comparing the performance of models augmented with

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 318–327
August 16, 2024. ©2024 Association for Computational Linguistics

UMLS knowledge with and without RAG, and against those without augmentation.

2 Related Work

2.1 Few-shot in-context learning

With the introduction of GPT models, there have been several works around few-shot in-context learning for clinical entity extraction where prompts guide information extraction in a contextually relevant manner (Agrawal et al., 2022; Hu et al., 2024; Shyr et al., 2024, Brown et al., 2020). Generative models can provide nuanced contextual understanding to extract clinical concepts, but cannot identify all domain-specific terminologies, especially in the clinical domain (Tang et al., 2023b). While recent language models have demonstrated improvement over prior language models (Guevara et al., 2024), there remains room for performance improvement.

2.2 Knowledge base-guided models

Previous research has explored the integration of knowledge bases to enhance information extraction tasks. (Sastre et al., 2020) proposed a Bi-LSTM model to identify drug-related information and integrate it into knowledge graph embeddings to evaluate drug identification accuracy. (Gilbert et al., 2024) addressed how knowledge bases complement language models for medical information identification tasks. Recently, a RAG model, Almanac, demonstrated significant performance improvements compared to the standard LLMs across various metrics (Zakka et al., 2024), further showing the benefits of access to domain-specific corpora for information extraction.

3 Methods

3.1 Overview of the Framework

Our approach leverages the context-capturing capability of GPT and knowledge-capturing capability of UMLS. UMLS contains a comprehensive list of more than 1 million biomedical concepts from over 100 source vocabularies. By using the concepts in the prompts in a few-shot learning setting, we attempt to improve GPT's ability to identify entities with the specified context that it may otherwise fail to extract independently. We map UMLS concepts to each text instance to create dynamic prompts unique to the specific context of the clinical text.The overview of the proposed framework is displayed in Figure 1.

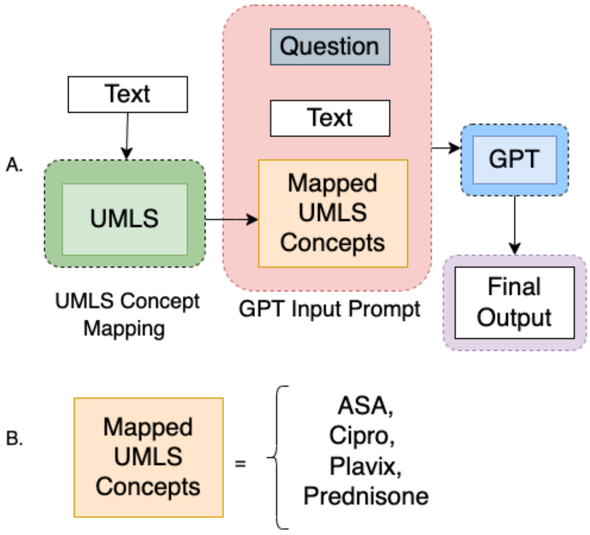
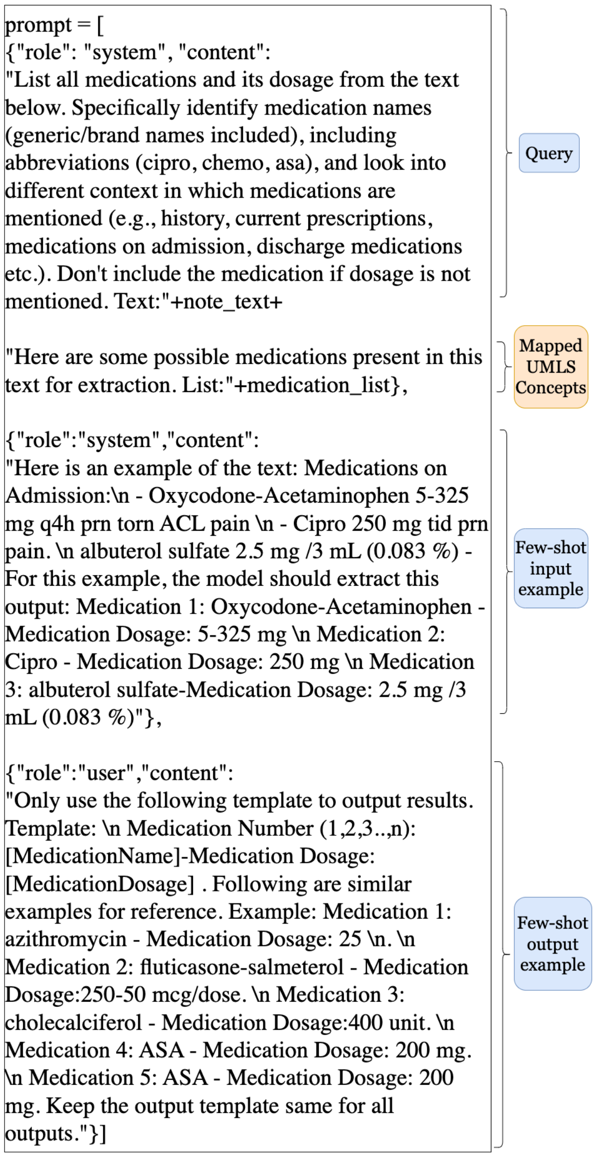


Figure 1: (A) Step-by-step approach to integrating UMLS and extracting relation pairs. (B) Example of UMLS concepts mapped from the text. Some of the concepts, such as Prednisone, are recognized by GPT, as they are concepts GPT model is inherently trained on. However, concepts such as ASA, Cipro, Plavix are not recognized by GPT; UMLS facilitates their recognition.

3.2 UMLS Integration in Large Language Model

UMLS Concept Mapping
We first map UMLS concepts from clinical text using MetaMap (Aronson, 2001). Given clinical text $X = \{x_1, x_2, \ldots, x_n\}$ where x_i represents the ith clinical text, we map $C_i = \{c_{i1}, c_{i2}, \ldots, c_{in}\}$, where C_i denote the set of concepts identified by MetaMap from x_i. n denotes the number of concepts identified from x_i. These concepts are extracted leveraging MetaMap's lexical parsing, syntactic analysis, semantic mapping, and concept mapping techniques.

Next, we filter the mapped concepts to include only those categorized as 'organic chemical', 'antibiotic', or 'pharmacologic substance' within the UMLS concept hierarchy as these groups contains the medications. For this work, we only target and filter medication-related concepts for augmentation and for further analysis. Let's denote the filtered set of concepts for the ith input clinical text x_i as $C_{filtered}$, such that $C_{filtered,i} = \{c_{ij} \in C_i | c_{ij} \in \text{filtered groups}\}$.



```
prompt = [
{"role": "system", "content":
"List all medications and its dosage from the text
below. Specifically identify medication names
(generic/brand names included), including
abbreviations (cipro, chemo, asa), and look into
different context in which medications are
mentioned (e.g., history, current prescriptions,
medications on admission, discharge medications
etc.). Don't include the medication if dosage is not
mentioned. Text:"+note_text+

"Here are some possible medications present in this
text for extraction. List:"+medication_list},

{"role":"system","content":
"Here is an example of the text: Medications on
Admission:\n - Oxycodone-Acetaminophen 5-325
mg q4h prn torn ACL pain \n - Cipro 250 mg tid prn
pain. \n albuterol sulfate 2.5 mg /3 mL (0.083 %) -
For this example, the model should extract this
output: Medication 1: Oxycodone-Acetaminophen -
Medication Dosage: 5-325 mg \n Medication 2:
Cipro - Medication Dosage: 250 mg \n Medication
3: albuterol sulfate-Medication Dosage: 2.5 mg /3
mL (0.083 %)"},

{"role":"user","content":
"Only use the following template to output results.
Template: \n Medication Number (1,2,3..,n):
[MedicationName]-Medication Dosage:
[MedicationDosage] . Following are similar
examples for reference. Example: Medication 1:
azithromycin - Medication Dosage: 25 \n. \n
Medication 2: fluticasone-salmeterol - Medication
Dosage:250-50 mcg/dose. \n Medication 3:
cholecalciferol - Medication Dosage:400 unit. \n
Medication 4: ASA - Medication Dosage: 200 mg.
\n Medication 5: ASA - Medication Dosage: 200
mg. Keep the output template same for all
outputs."}]
```

Figure 2: An example of a prompt used to extract dosage information from the text using the UMLS concepts. The 'note_text' represents each text instance from ADE or n2c2 corpus. The 'medication_list' represents the UMLS concepts extracted from MetaMap.

Prompt Strategy and Large Language Model Implementation

Next, we prompt the GPT model to extract entity-relation pairs from the text, leveraging the mapped UMLS concepts from MetaMap, and employing a few-shot prompt strategy. Let P_i represent the prompt generated for each input text x_i, incorporating the relevant UMLS concepts $C_{filtered,i}$. The final prompt P_i is constructed as the concatenation of the initial prompt and the set of UMLS concepts, i.e., $P_i = \text{Concat}(P, C_{filtered,i})$. We use OpenAI's GPT-4-32k (Version 0613) and GPT-3.5-turbo (Version 0301) via HIPPA-compliant Microsoft Azure's OpenAI REST API[1]endpoint. A sample prompt and hyperparameters used by the models for this task are available in Figure 2 and A.2 respectively. As our goal for the project was not to explore different prompting strategies, we tested a few prompts and selected the prompt that generated more specific result. We used the same format for all relation pairs replacing only the entity type for every run.

Retrieval Augmented Generation

We also explored another approach-RAG to leverage UMLS in a language model, which is a more conventional method involving the use of external data. RAG was chosen for its potential to enhance the generation process by incorporating domain-specific knowledge from sources like the UMLS knowledge base. Appendix A.4 includes details on our RAG implementation.

3.3 Datasets Description

We used the n2c2 and ADE datasets for our experiments.

n2c2 Dataset

We used a curated National NLP Clinical Challenges (n2c2) dataset (Henry et al., 2019) consisting of 303 deidentified discharge summaries obtained from the MIMIC-III (Medical Information Mart for Intensive Care-III) critical care database (**Table 1A**) (Johnson et al., 2016). The data also contained annotations of medication-related entities and their relationship to other entities. Annotations conducted by 3 subject matter experts served as a gold standard to evaluate model performance.

ADE Dataset

The Adverse Drug Events (ADE) dataset annotated by 5 individuals consists of MEDLINE[2] case reports with information on medications, dosages and adverse effects associated with the medications (Gurulingappa et al., 2012) (**Table 1B**). It also contains relationships between medications, dosages, and adverse effects. For our experiments, we used the second version of the dataset downloaded from

[1] https://learn.microsoft.com/en-us/azure/ai-services/openai/quickstart?tabs=command-line%2Cpython-new&pivots=programming-language-python

[2] https://www.nlm.nih.gov/medline/medline_home.html

Huggingface[3].

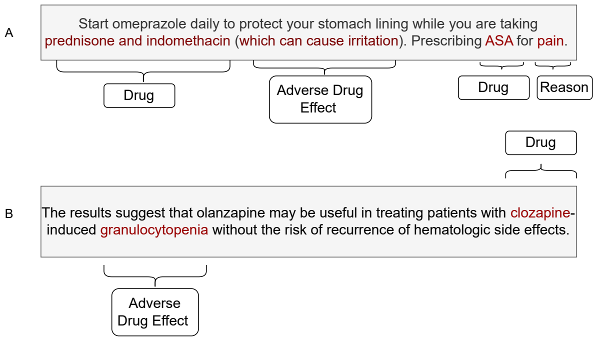

A
Start omeprazole daily to protect your stomach lining while you are taking prednisone and indomethacin (which can cause irritation). Prescribing ASA for pain.
Drug
Adverse Drug Effect
Drug
Reason
Drug
B
The results suggest that olanzapine may be useful in treating patients with clozapine-induced granulocytopenia without the risk of recurrence of hematologic side effects.
Adverse Drug Effect

Figure 3: Sample text of discharge summaries in the (A) n2c2 dataset and (B) ADE Corpus. The text highlighted in red are the targeted entities for extraction

Table 1: Statistics on the relation pairs in the (A) n2c2 dataset and the (B) ADE dataset

A. n2c2 Dataset	
Entity-Entity Relation	**Total instances**
Strength-Drug	13338
Duration-Drug	643
Route-Drug	11038
Form-Drug	6636
ADE-Drug	2214
Dosage-Drug	4207
Reason-Drug	5160
Frequency-Drug	6288

B. ADE Dataset	
Entity-Entity Relation	**Total instances**
Drug-ADE	6821
Drug-Dosage	279

4 Results

4.1 Experimental Setup

We evaluated two generative models, GPT-4-32k and GPT-3.5-turbo, with and without UMLS integration, and the RAG model to access the quality of generated outputs. All models were evaluated against the gold-standard annotations using precision, recall, and micro-F1 score.

4.2 Dataset

We identified 8 different entity-entity relation pairs within the n2c2 dataset, and 2 entity-entity relation pairs within the ADE corpus, each with varying instances of the relation pairs (**Table 1, Figure 3**). Token length distributions of the text, and example of individual entities in n2c2 and ADE dataset are available in A.1, A.3.

4.3 Performance Results

Results on n2c2 and ADE Dataset

Our results suggests that integrating prior knowledge from UMLS in the prompts have significant performance improvement as demonstrated by the higher average F-1 scores across both n2c2 and ADE datasets (**Table 4**). The reported results are average across all entity-entity relation pairs across models and for 2 datasets. GPT-4-32k model with UMLS show 4% improvement of F-1 score on the n2c2 dataset, and 12% improvement on the ADE dataset from the F-1 score of GPT-4-32k model without knowledge integration. For every entity-entity relation pairs, there was a performance improvement by a few percentages for both models and across both datasets. Additional detailed results for each entity-entity relation pairs can be found in Appendices A.6 through A.11.

Upon a closer look at the results, we identified that prompts with UMLS resulted in additional concepts verifying that UMLS is able to identify medications from the text that GPT may not identify independently(Appendix A.5).

Comparison with Retrieval Augmented Generation

RAG model and GPT-3.5-turbo had low F-1 score and it improved with UMLS for both models, but it did not have higher score compared to the GPT-4-32k+UMLS.

We observed performance variations across entity-entity relation pairs with retrieval augmented generation (A.8, A.11). While some entity pairs showed performance enhancements, others did not show significant improvements. This discrepancy might arise from the limitations of RAG, particularly its inability to utilize entire UMLS thesaurarus in the generation process. Since UMLS data is partitioned into chunks for indexing and embedding and embedding models can only take 8192 tokens per index, some concepts may not be in the top-k extracted documents used for generation, potentially limiting the scope of augmentation and its impact on final relation pairs. Further experiments are required to confirm this hypothesis.

[3] https://huggingface.co/datasets/ade_corpus_v2

Table 2: Comparison of models on n2c2 Dataset and ADE Corpus. The reported results are micro-averaged precision, recall, and F-1 scores across all entity-entity relation pairs within the datasets.

Model	n2c2 Dataset			ADE Corpus		
	Precision	**Recall**	**F1**	**Precision**	**Recall**	**F1**
GPT-3.5-turbo	0.73	0.74	0.73	0.625	0.57	0.596
GPT-3.5-turbo + UMLS	0.77	0.77	0.77	0.83	0.70	0.75
RAG w/ GPT-3.5-turbo	0.73	0.74	0.74	0.65	0.63	0.64
GPT-4-32k	0.75	0.76	0.76	**1.0**	0.70	0.82
GPT-4-32k + UMLS	**0.79**	**0.80**	**0.80**	**1.0**	**0.89**	**0.94**
RAG w/ GPT-4-32k	0.77	0.76	0.76	**1.0**	0.74	0.85

5 Discussion and Conclusion

Our study highlights the significance of merging the strengths of domain-specific knowledge bases, such as UMLS, with the contextual understanding capabilities of LLMs, such as GPT. Our hybrid approach, integrating mapped UMLS concepts with GPT, shows improvement in the model's ability to identify specific entities not inherently within its training data.

Our results on entity and relation extraction task indicated that leveraging mapped UMLS concepts as additional guidance to the GPT model, helps create focused and unique prompts that significantly enhances GPT's performance. This approach proves more effective than the standard RAG technique.

In conclusion, the ability to generate tailored prompts based on UMLS concepts offers a promising avenue for improving accuracy and relevance of extracted entities, ultimately enhancing the utility of LLMs in biomedical text analysis tasks.

6 Limitations and Future Work

While our framework has shown significant improvements, we acknowledge several limitations in this study. Firstly, our work focused solely on medication concepts, which may restrict the generalizability of our findings to other concepts. However, our approach is adaptable to incorporate additional UMLS entities through prompt adjustments. Future research will explore harnessing UMLS's rich semantic metadata to leverage additional concept relationships, enabling the extraction of a broader spectrum of entity groups beyond medications.

Secondly, our comparison was limited to two generative models, GPT-4-32k and GPT-3.5-turbo. Though they have good performance, we have not included recent models that could have comparable performance. Future work will explore additional models, such as BioGPT, and LAMA for comprehensive comparison and evaluation. This expanded comparison will provide a more nuanced understanding of the performance and capabilities of various generative models in relation to UMLS integration and RAG techniques.

These future tasks will advance our understanding of the role of domain-specific knowledge in enhancing LLM capabilities and facilitating more effective clinical information extraction.

7 Ethics Statement

IRB approval was not required for this task. To input our text data into the language models, we use Microsoft's Azure OpenAI REST API Service within the Washington University tenant to access OpenAI's language models . We are on a HIPPA-compliant subscription and exempted from content filtering, data review and human review for our use of the Azure OpenAI service.

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1998–2022.

A R Aronson. 2001. Effective mapping of biomedical text to the umls metathesaurus: the metamap program. *Proc AMIA Symp.*

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *AAAI*, 32:D267–D270.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, page 4171–4186.

Stephen Gilbert, Jakob Nikolas Kather, and Aidan Hogan. 2024. Augmented non-hallucinating large language models as medical information curators. *npj Digit. Med*, 7(100).

Marco Guevara, Shan Chen, Spencer Thomas, Tafadzwa L. Chaunzwa, Idalid Franco, Benjamin H. Kann, Shalini Moningi, Jack M. Qian, Madeleine Goldstein, Susan Harper, Hugo J. W. L. Aerts, Paul J. Catalano, Guergana K. Savova, Raymond H. Mak, and Danielle S. Bitterman. 2024. Large language models to identify social determinants of health in electronic health records. *Nature*, 7(6).

Harsha Gurulingappa, Abdul Mateen Rajput, Angus Roberts, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. 2012. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *Journal of Biomedical Informatics*, 45(5):885–892.

Sam Henry, Kevin Buchan, Michele Filannino, Amber Stubbs, and Ozlem Uzuner. 2019. 2018 n2c2 shared task on adverse drug events and medication extraction in electronic health records. *Journal of the American Medical Informatics Association*, 27(1):3–12.

Yan Hu, Qingyu Chen, Jingcheng Du, Xueqing Peng, Vipina Kuttichi Keloth, Xu Zuo, Yujia Zhou, Zehan Li, Xiaoqian Jiang, Zhiyong Lu, Kirk Roberts, and Hua Xu. 2024. Improving large language models for clinical named entity recognition via prompt engineering. *Journal of the American Medical Informatics Association*, pages 1–10.

Alistair E.W. Johnson, Tom J. Pollard, Lu Shen, Li wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. 2016. Mimic-iii, a freely accessible critical care database. *Journal of the American Medical Informatics Association*, 3(160035).

Javier Sastre, Faisal Zaman, Noirin Duggan, Caitlin McDonagh, and Paul Walsh. 2020. A deep learning knowledge graph approach to drug labelling. *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 32:2513–2521.

Cathy Shyr, Yan Hu, Lisa Bastarache, Alex Cheng, Rizwan Hamid, Paul Harris, and Hua Xu. 2024. Identifying and extracting rare disease phenotypes with large language models. *Journal of the American Medical Informatics Association*, 8:438–461.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Abubakr Babiker, Nathanael Schärli, Aakanksha Chowdhery, Philip Mansfield, Dina Demner-Fushman, Blaise Agüera y Arcas, Dale Webster, Greg S. Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. 2023. Large language models encode clinical knowledge. *Nature*, 620:172–180.

Liyan Tang, Zhaoyi Sun, Betina Idnay, Jordan G. Nestor, Ali Soroush, Pierre A. Elias, Ziyang Xu, Ying Ding, Greg Durrett, Justin F. Rousseau, Chunhua Weng, and Yifan Peng. 2023a. Evaluating large language models on medical evidence summarization. *Nature*, 6(158).

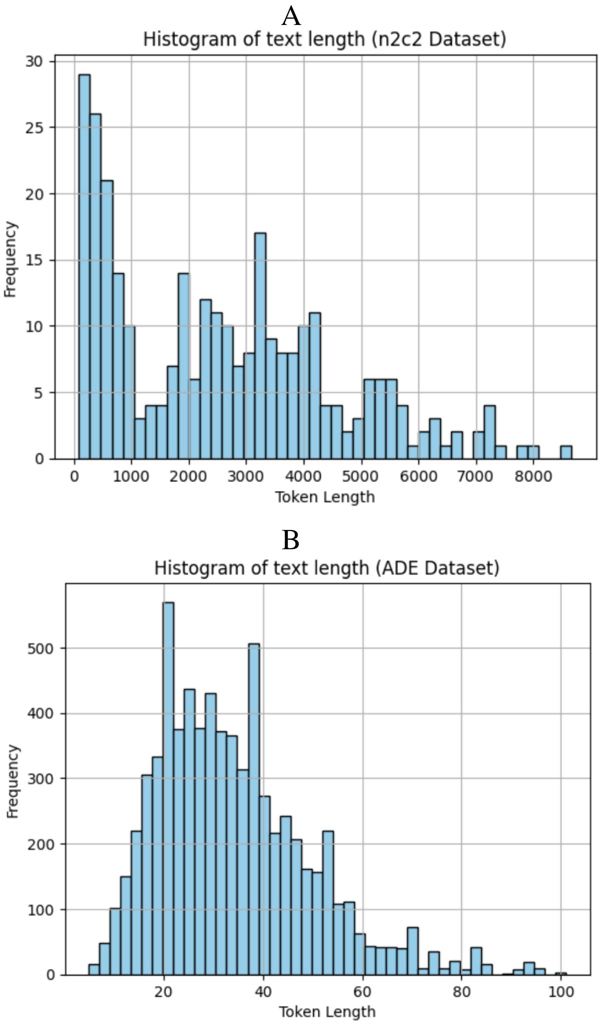
Xiangru Tang, Anni Zou, Zhuosheng Zhang, Ziming Li, Yilun Zhao, Xingyao Zhang, Arman Cohan, and Mark Gerstein. 2023b. Medagents: Large language models as collaborators for zero-shot medical reasoning. *Computing Research Repository*, arXiv:2311.10537. Version 3.

Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. 2022. An empirical study of gpt-3 for few-shot knowledge-based vqa. *AAAI*, 36:3081–3089.

Cyril Zakka, Rohan Shad, Akash Chaurasia, Alex R. Dalal, Jennifer L. Kim, Michael Moor, Robyn Fong, and William Hiesinger. 2024. Almanac — retrieval-augmented language models for clinical medicine. *npj Digit. Med*, 1(2).

A Appendix

A.1 Token length of the text in (A) n2c2 and (B) ADE dataset



A.2 Hyperparameters for the GPT models

Hyperparameters	Value
Tokenization and Context Window	200 tokens
Temperature (Randomness of the model output)	0
Top p (Top-K Sampling Technique)	0.95
Presence Penalty (Penalty to discourage model from generating responses that contain certain specified tokens)	-1.0

A.3 Example of the individual entities within the n2c2 and ADE dataset

Table 3: Example of the individual entities within the n2c2 and ADE dataset

Entities	Examples
Drug	Morphine, ibuprofen, antibiotics (abx), chemotherapy (carboplatin)
ADE/Reason	Nausea, rash, seizures, vitamin K deficiency
Strength	10 mg, 60 mg
Form	Capsule, syringe, tablet, topical (apply topical)
Dosage	60 mg/0.6 mL
Frequency	Daily, twice a day, Q4H (every 4 hours)
Route	Transfusion, oral, intravenous (IV)
Duration	For 10 days, 2 cycles, for a week

A.4 Retrieval Augmented Generation

Method:
1. Split UMLS data (MRCONSO.RRF[4]) into manageable chunks (8192 tokens) to facilitate processing. MRCONSO.RRF file contains the UMLS concepts.
2. Generate embeddings for each chunk, capturing its semantic represetations
2. Store the embeddings in a vector database for efficient retrieval
3. Compare each prompt with the stored data in the vector database.
4. Extract the top 30 results with the highest similarity scores between the query and the UMLS data.
5. Concatenate the retrieved results with the prompt to generate the final extraction output.

A.5 Qualitative Results

Table 4: Some of the qualitative results for the Strength-Drug Pair. (A) Without UMLS integration. (B) With UMLS integration

	Examples
A	[('aspirin', '81 mg') ('atorvastatin', '20 mg'), ('amiodarone', '200 mg'), ('metoprolol tartrate', '50 mg'), ('spironolactone', '25 mg'), ('acetaminophen', '325 mg'), ('ranitidine HCl', '150 mg'), ('prednisone', '60 mg')]
B	[('aspirin', '81 mg'), ('atorvastatin', '20 mg'), ('amiodarone', '200 mg'), ('metoprolol tartrate', '50 mg'), ('spironolactone', '25 mg'), ('acetaminophen', '325 mg'), ('ranitidine HCl', '150 mg'), ('prednisone', '60 mg'), **('Plavix', '75 mg')**, **('ASA', '325')**, **('Cipro', '250 mg')**]

[4] https://www.ncbi.nlm.nih.gov/books/NBK9685/table/ch03.T.concept_names_and_sources_file_mr/

A.6 Comparison of GPT-3.5-turbo for all Entity-Entity Relation pairs with and without UMLS Integration for the n2c2 dataset

Entity-Entity	GPT-3.5-turbo			GPT-3.5-turbo+UMLS		
	P	R	Micro F-1	P	R	F-1
Dosage-Drug	0.75	0.75	0.75	0.80	0.80	0.80
Duration-Drug	0.76	0.76	0.76	0.81	0.81	0.81
Route-Drug	0.74	0.73	0.73	0.76	0.75	0.75
Form-Drug	0.72	0.73	0.72	0.75	0.76	0.75
ADE-Drug	0.69	0.71	0.70	0.74	0.75	0.75
Reason-Drug	0.73	0.74	0.74	0.76	0.77	0.777
Frequency-Drug	0.73	0.74	0.73	0.75	0.76	0.77
Average	0.73	0.74	0.73	0.77	0.77	0.77

A.7 Comparison of GPT-4-32k for all Entity-Entity Relation pairs without UMLS Integration for the n2c2 dataset

Entity-Entity	GPT-4-32k			GPT-4-32k+UMLS		
	P	R	Micro F-1	P	R	F-1
Dosage-Drug	0.77	0.77	0.77	0.82	0.82	0.82
Duration-Drug	0.78	0.77	0.78	0.83	0.82	0.83
Route-Drug	0.79	0.77	0.78	0.81	0.78	0.79
Form-Drug	0.74	0.76	0.74	0.77	0.79	0.77
ADE-Drug	0.69	0.73	0.71	0.75	0.78	0.77
Reason-Drug	0.74	0.75	0.735	0.77	0.78	0.76
Frequency-Drug	0.78	0.77	0.78	0.80	0.79	0.79
Average	0.75	0.76	0.76	0.79	0.79	0.79

A.8 Comparison of Models for all Entity-Entity Relation pairs with UMLS for RAG on the n2c2 dataset

Entity-Entity	GPT-4-32k			GPT-3.5-turbo		
	P	R	F-1	P	R	F-1
Dosage-Drug	0.77	0.77	0.77	0.75	0.75	0.75
Duration-Drug	0.79	0.78	0.78	0.76	0.77	0.77
Route-Drug	0.79	0.77	0.78	0.74	0.73	0.73
Form-Drug	0.74	0.73	0.74	0.73	0.74	0.74
ADE-Drug	0.72	0.73	0.72	0.70	0.70	0.70
Reason-Drug	0.76	0.76	0.76	0.75	0.78	0.76
Frequency-Drug	0.81	0.80	0.80	0.70	0.71	0.71
Average	0.77	0.76	0.76	0.73	0.74	0.74

A.9 Comparison of GPT-4-32k for all Entity-Entity Relation pairs with and without UMLS on the ADE dataset

Entity-Entity	**GPT-4-32k**			**GPT-4-32k+UMLS**		
	P	**R**	**F-1**	**P**	**R**	**F-1**
Dosage-Drug	1.0	0.66	0.795	1.00	0.85	0.91
ADE-Drug	1.0	0.73	0.84	1.00	0.93	0.97
Average	1.0	0.70	0.82	1.0	0.89	0.94

A.10 Comparison of GPT-3.5-turbo for all Entity-Entity Relation pairs with and without UMLS on the ADE dataset

Entity-Entity	**GPT-3.5-turbo**			**GPT-3.5-turbo+UMLS**		
	P	**R**	**F-1**	**P**	**R**	**F-1**
ADE-Drug	0.57	0.53	0.55	0.60	0.65	0.62
Dosage-Drug	0.68	0.61	0.64	0.70	0.75	0.72
Average	0.625	0.57	0.596	0.83	0.70	0.75

A.11 Comparison of the models for all Entity-Entity Relation pairs with UMLS for RAG on the ADE dataset

Entity-Entity	**GPT-4-32k**			**GPT-3.5-turbo**		
	P	**R**	**F-1**	**P**	**R**	**F-1**
ADE-Drug	1.0	0.73	0.84	0.62	0.61	0.60
Dosage-Drug	1.0	0.75	0.86	0.68	0.65	0.66
Average	1.0	0.74	0.85	0.65	0.63	0.64

BiCAL: Bi-directional Contrastive Active Learning for Clinical Report Generation

Tianyi Wu[1], Jingqing Zhang[1, 2], Wenjia Bai[1], Kai Sun[1]
[1]Imperial College London [2]Pangaea Data
[1]{andrew.wu22, jingqing.zhang15, w.bai, k.sun}@imperial.ac.uk
[2]jzhang@pangaeadata.ai

Abstract

State-of-the-art performance by large pre-trained models in computer vision (CV) and natural language processing (NLP) suggests their potential for domain-specific tasks. However, training these models requires vast amounts of labelled data, a challenge in many domains due to the cost and expertise required for data labelling. Active Learning (AL) can mitigate this by selecting minimal yet informative data for model training. While AL has been mainly applied to single-modal tasks in the fields of NLP and CV, its application in multi-modal tasks remains underexplored. In this work, we proposed a novel AL strategy, **Bi**directional **C**ontrastive **A**ctive **L**earning strategy (BiCAL), that used both image and text latent spaces to identify contrastive samples to select batches to query for labels. BiCAL was robust to class imbalance data problems by its design, which is a problem that is commonly seen in training domain-specific models. We assessed BiCAL's performance in domain-specific learning on the clinical report generation tasks from chest X-ray images. Our experiments showed that BiCAL outperforms State-of-the-art methods in clinical efficacy metrics, improving recall by 2.4% and F1 score by 9.5%, showcasing its effectiveness in actively training domain-specific multi-modal models.

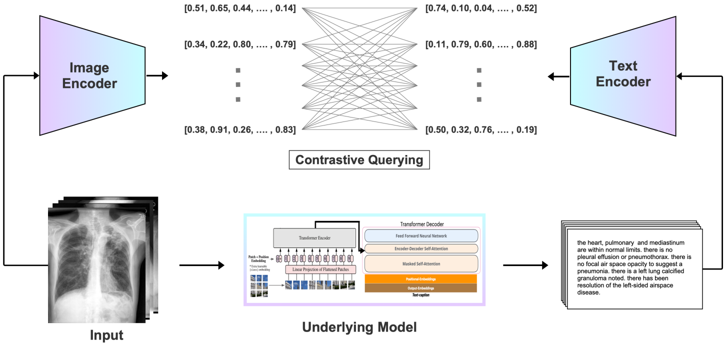

[0.51, 0.65, 0.44, ... , 0.14]
[0.34, 0.22, 0.80, ... , 0.79]
[0.38, 0.91, 0.26, ... , 0.83]
[0.74, 0.10, 0.04, ... , 0.52]
[0.11, 0.79, 0.60, ... , 0.88]
[0.50, 0.32, 0.76, ... , 0.19]
Image Encoder
Text Encoder
Contrastive Querying
Transformer Encoder
Transformer Decoder
Feed Forward Neural Network
Encoder-Decoder Self-Attention
Masked Self-Attention
Patch + Position Embedding
Linear Projection of Flattened Patches
Positional-Embeddings
Output-Embeddings
Text-caption
the heart, pulmonary and mediastinum are within normal limits. there is no pleural effusion or pneumothorax. there is no focal air space opacity to suggest a pneumonia. there is a left lung calcified granuloma noted. there has been resolution of the left-sided airspace disease.
Input
Underlying Model

Figure 1: Flowchart of the querying process of BiCAL: Image is passed to imaged encoder to obtain image embeddings, and the underlying training model to generate reports. Reports generated are passed to a text encoder to generate text embeddings. Together two embeddings are compared and the contrastiveness of each data point is calculated and queried. Refer detail to Algorithm 1.

1 Introduction

Active Learning (AL) is a branch of machine learning that aims to select a small set of the most informative data to annotate for model training (Settles, 2009). This technique allows the model to achieve optimal performance while lowering the cost of annotation. Moreover, by actively selecting data to train on, a model trained under active learning can sometimes surpass the performance that is trained on the full dataset. AL has shown its great potential in the field of natural language processing (NLP) (Shelmanov et al., 2021; Dor et al., 2020; Shen et al., 2017; Margatina et al., 2021a) and computer vision (CV) (Slade and Branson, 2022; Takezoe et al., 2023). However, relatively few have explored the application of active learning in a multi-modal setting.

In addition, as the capabilities of general large-pretrained models arise (Bai et al., 2023; OpenAI, 2023), a rising interest has been seen in fine-tuning them to become domain-specific models. However, when training models in specific domains, obtaining quality labelled data is challenging due to the domain expertise required for accurate annotation, which is costly in both time and money. This motivates us to explore active learning's application in the domain-specific setting. We identify that in domain-specific active learning, there exists one key challenge – class imbalance in datasets is often seen in domain-specific settings, existing AL methods struggle to actively select samples that have less population but may be more important – in medicine, common (healthy) samples often out populate rare (unhealthy) samples. Models trained

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 328–341
August 16, 2024. ©2024 Association for Computational Linguistics

under such active learning strategies converge on the commonly seen samples and perform poorly in identifying rare sickness cases.

In this study, we introduce a novel AL strategy **Bi**directional **C**ontrastive **A**ctive **L**earning strategy (BiCAL) that is tailored to address the challenge in domain-specific active learning. We assess BiCAL and other established AL methods on clinical report generation from chest X-ray images. Our key contributions are:

1. We propose a novel AL strategy BiCAL that is able to select rare but important cases inherently to be robust against the class imbalance limitations, which is a common problem in clinical setting.
2. We present an in-depth analysis of existing AL strategies for multi-modal task – clinical report generation.

2 Related Work

This section provides the background of our proposed AL strategy BiCAL. We first formalize the active learning problem under the image-to-text generation task and set up the notation for the rest of the paper. Given a model $\mathcal{M}$, unlabelled image data pool X_{pool}. We denote an unlabelled input image as $x \in X_{pool}$, and the labelled text report as $y \in Y$, where $y = (y^1, ..., y^n)$, n is the number of tokens in the generated report. We define the labelled data pool X_{label} to contain image-report pairs, where $X_{\text{label}} \cap X_{\text{pool}} = \emptyset$. The whole data pool is $X_{all} := X_{label} \cup X_{pool}$. The model is parameterized by vector w, as follows:

$$\mathcal{M} = p_w(y \mid x) = p_w(y^1, ..., y^n \mid x) \qquad (1)$$

An acquisition function representing the query heuristic in the AL setting is denoted as $a(x, \mathcal{M})$. At each active learning iteration, we acquire the label of a batch Q of b number of unlabelled instances from X_{pool} and add to the labelled data pool X_{label} using $a(x, \mathcal{M})$. The updated labelled data pool X_{label} is used to train the underlying model every iteration. This process iterates until a predefined budget $\mathcal{B}$ is depleted. Sampling from the pool is determined by the acquisition function as follows :

$$x^* = \text{argmax}_{x \in X_{\text{pool}}} a(x, \mathcal{M}) \qquad (2)$$

2.1 Uncertainty-based and Diversity-based Active Learning

Uncertainty-based AL strategies often use a heuristic that can measure the model's uncertainty toward unlabelled data and choose the unlabelled data with the highest uncertainty (Lewis, 1995; Wang et al., 2019; Shannon, 2001). Gal et al. (2017) demonstrated the idea of measuring model uncertainty by combining Bayesian Active Learning by Disagreement (BALD) (Houlsby et al., 2011) with Bayesian formulation of Neural Networks such as Bayesian by Backprop (Blundell et al., 2015) and MC dropout (Gal and Ghahramani, 2016). However, uncertainty-based active learning typically depends on the underlying training model's predictions for uncertainty measurements. This dependence results in the "cold-start" problem (Yuan et al., 2020; Ash and Adams, 2020), where these methods are ineffective early in training due to the initial model's naivety.

On the other side, diversity-based Active Learning aims to select a subset of the data that can best represent the whole dataset, such that the model achieves similar performance to full-tuning when trained on the selected subset. There has been much previous work in this stream of designing AL strategies (Kim et al., 2006; Citovsky et al., 2021; Sener and Savarese, 2018).

2.2 Hybrid Active Learning

There have also been some hybrid AL methods that combine diversity and uncertainty in their design (Ash et al., 2019; Yuan et al., 2020). Approaches that infuse reinforcement learning into AL strategies which learn the selection heuristic from scratch were also seen (Fang et al., 2017; Liu et al., 2018; Vu et al., 2019). There has been close work on active learning to ours in natural language generation and abstractive text summarization, however, they focused on the single modal generation task (Tsvigun et al., 2023; Gidiotis and Tsoumakas, 2021a; Perlitz et al., 2023; Gidiotis and Tsoumakas, 2021b).

The closest work to ours is Contrastive Active Learning (CAL) proposed by (Margatina et al., 2021b). They hypothesized that if two data points are close in the underlying model feature space but result in very different underlying model predictive likelihood, then they may be lying on the model's decision boundary and therefore are a good candidate to query. CAL uses K-Nearest Neighbors (KNN) (Cover and Hart, 1967) to find and record the top k neighbouring points by their model representation encodings from the input. Then it computes the KL divergence (Kullback and Leibler, 1951) between the model's output probability of each unlabelled instance with their recorded k neighbours. The contrastive score of each unlabelled instance is then calculated by the average of all KL-divergence values of the neighbours. Ultimately, the data point with the highest contrastive score is selected to be queried.

3 Bidirectional Contrastive Active Learning

We identify the following limitations that existing AL methods have when training models in a clinical setting. In clinical settings, data for healthy or common sicknesses is often seen, while unhealthy or rare sicknesses are rare in the population, leading to an imbalanced dataset. This leads to models trained on such datasets that can converge easily on the common cases, and have poor performance on rare but important cases. Previous AL methods have not yet addressed this problem, as they are not able to explicitly identify important cases within the dataset automatically. The original CAL would identify two data points are neighbours if two data points have the same sickness, and if the model predicts differently for the two data points, they are considered as 'contrastive' and queried. Such a heuristic cannot locate the positive (unhealthy) cases efficiently, because negative (healthy) neighbour pairs would outweigh the positive (unhealthy) neighbour pair in the population, leading to the sampling process suffering from class imbalance and queries too many negative instances. Therefore, models trained using CAL achieve a bad performance in clinical efficacy and recalling positive cases, as revealed by our experiments in Table 1.

3.1 BiCAL Algorithm

BiCAL is robust to class imbalance datasets by its design and can automatically select rare but valuable cases within a dataset for the model to learn. This is done by bi-directionally augmenting the contrastive definition and measuring the contrastiveness in pre-trained embedding space, empowering the algorithm to select rare samples in domain datasets inherently.

We redefine two types of contrastive samples. For BiCAL, contrastive examples have to satisfy one of the following definitions:

1. Two data points with **similar** pre-trained embeddings but **different** pre-trained embeddings of their model generation outputs.
2. Two data points with **different** pre-trained embeddings but **similar** pre-trained embeddings of their model generation outputs.

The intuition behind the second augmented definition is that common cases and rare cases will most likely have the most different representations of each other within the dataset. Therefore, if a model generates similar outputs for two data points that have different representations, this means it is highly possible that at least one rare sample is within the two data points, and the current model hasn't trained enough on at least one of the two data points. Hence by augmenting the contrastive definition in BiCAL, we have increased the chance of querying a rare case, compared to CAL. Moreover, by leveraging pre-trained encoders, we isolate the underlying model in generating the uncertainty measure, alleviating the Cold-Start Learning problem (Yuan et al., 2020; Ash and Adams, 2020) – in the initial stage of training, the underlying model is naive due to the absence of domain knowledge, if we use the underlying model's encoder to generate uncertainty measure it would result in a decrease in the ineffectiveness of such uncertainty-based AL strategies.

Formally, each data point x_i should obtain k number of similar neighbours X_{close} and k number of dissimilar neighbours X_{far}.

$$\begin{aligned} X_{close} &:= & f(\Phi(x_i), \Phi(x_j)) < \epsilon \\ X_{far} &:= & f(\Phi(x_i), \Phi(x_j)) > \gamma \end{aligned} \tag{3}$$

For the first contrastive sample, the data point should satisfy the following condition:

$$f(\Omega(\mathcal{M}(x_i)), \Omega(\mathcal{M}(x^m_{close}))) > \gamma \tag{4}$$

For the second contrastive sample, the data point should satisfy the following conditions:

$$f(\Omega(\mathcal{M}(x_i)), \Omega(\mathcal{M}(x^m_{far}))) < \epsilon \tag{5}$$

Where $\Phi(.) \in \mathbb{R}^{d'}$ is a selected pre-trained image encoder that maps input x_i and x_j to its feature space. $\Omega(.) \in \mathbb{R}^{d''}$ is the selected pre-trained text encoder that maps the predicted output of underlying

Algorithm 1 Single iteration of BiCAL

```
Input: all data $X_{all}$, unlabeled data $X_{pool}$, acquisition size $b$, model $\mathcal{M}$, number of neighbours $k$, distance metric function $f(.)$, pre-trained image
       (encoding) function $\Phi(.)$, pre-trained text (encoding) function $\Omega(.)$, contrastive ratio $c \in [0,1]$, Total number of unlabelled data $N$, .
1  $S_{close} := \emptyset$ ; $S_{far} := \emptyset$
2  for $i$ in $1, \ldots, N$ do
3      $d_j \leftarrow f\Big(\Phi(x_i), \Phi(x_j)\Big)$                              ▷ $x_j \in X_{all}, j = 1, ..., N$
4      $X_{close} \leftarrow$ Select k number of x$\in X_{all}$ with lowest d$_j$     ▷ $X_{close} = \{x^1_{close}, ..., x^k_{close}\}$; $j \neq i$
5      $X_{far} \leftarrow$ Select k number of x$\in X_{all}$ with highest d$_j$     ▷ $X_{far} = \{x^1_{far}, ..., x^k_{far}\}$
6      $\hat{Y}_{close} \leftarrow \mathcal{M}(X_{close})$
7      $\hat{Y}_{far} \leftarrow \mathcal{M}(X_{far})$
8      $\hat{y}_i \leftarrow \mathcal{M}(x_i)$
9      $s^i_{close} \leftarrow \frac{1}{k} \sum_{m=1}^{k} f\Big(\Omega(\hat{y}_i), \Omega(\hat{y}^m_{close})\Big)$
10     $s^i_{far} \leftarrow \frac{1}{k} \sum_{m=1}^{k} f\Big(\Omega(\hat{y}_i), \Omega(\hat{y}^m_{far})\Big)$
11     $S_{close} := S_{close} \cup \{s^i_{close}\}$ ; $S_{far} := S_{far} \cup \{s^i_{far}\}$
12 end
13 $Q_1 \leftarrow$ Select $b \times c$ number of x $\in X_{pool}$ with the highest $s_{close}$          ▷ $s_{close} \in S_{close}$
14 $Q_2 \leftarrow$ Select $b \times (1-c)$ number of x $\in X_{pool}$ with the lowest $s_{far}$        ▷ $s_{far} \in S_{far}$
Output: $Q_1 \cup Q_2$
```

model $\hat{y}_i$ to its feature space. $f(.)$ is a distance metric, such as Euclidean distance or cosine similarity. ϵ and γ represent the threshold for a very small and a very large distance value respectively, although in practice we adopt ranking instead of using a threshold. $\mathcal{M}(.)$ is the underlying training model of the active learning loop, such that $\hat{y}_i \leftarrow \mathcal{M}(x_i)$. We detail the single iteration of BiCAL's algorithm as follows:

Compute Neighbours We use the encoding function from the pre-trained model $\Phi(.)$ to map all the data points to its pre-trained embedding space. For each unlabelled instance x_i, we use cosine similarity $f(.)$ to measure the distances between the embeddings of x_i and all the other data points in the X_{all} (line 3). We record x_i's nearest (top k) and furthest (bottom k) neighbours in the embedding space by the distance calculated (lines 4-5).

Compute Contrastive Scores The unlabelled instance x_i and all its neighbours X_{close} and X_{far} will be passed to the underlying model $\mathcal{M}$ to generate their text outputs $\hat{y}$ (lines 6-8). The generated text from the model is then encoded by the selected pre-trained language model $\Omega(.)$ to obtain text embedding of the generated text. Using these embeddings, we can calculate two different contrastive scores for the unlabelled instance x_i (lines 9-10). The first contrastive score s^i_{close} is calculated by the average distance between the embedding of generated output of the unlabelled instances with their nearest neighbours, and the second one s^i_{far} is calculated with its furthest neighbours.

Query Two Contrastive Batches For each unlabelled instance x_i, we obtain two lists of contrastive scores S_{close} and S_{far}. We select the unlabelled instances using the two contrastive scores separately. For S_{close}, we select the top $b \times c$ number of instances, where b is the total intended batch size for query, and c is a hyperparameter "contrastive ratio" that controls the ratios of samples sampled from the two contrastive definitions. This gives us a batch of instances Q_1 of the first contrastive definition (line 13). For S_{far}, we select the bottom $b \times (1-c)$ number of instances. This gives us a batch of instances Q_2 of the second contrastive definition (line 12). Ultimately, two batches Q_1 and Q_2 combines to give the output of BiCAL.

4 Experiment Settings

We assess BiCAL and other established AL methods' performance in training general multi-modal models to specify on the task of clinical report generation from chest X-ray images. In every active learning loop, the underlying model denoted as $\mathcal{M}$, was fine-tuned twice on the labelled pool X_{label}. Subsequently, we evaluated the model on the test dataset using various NLG metrics. Each experiment was run in 3 folds with different random seeds, each fold containing 10 active learning iterations, where 100 data points were queried per iteration, i.e. 1000 data points were queried in total. This choice of 1000 data points reflects real-world scenarios where active learning is applied when labelled data is not available. Our goal was to examine the efficiency and performance of active learning methods under constrained labelling budgets in a medical setting. In real-world AL scenarios, labelling a large size of unlabelled data is often impractical due to the significant expertise labelling effort required. Therefore, 1000 data points were deemed sufficient to assess the performance of the AL methods in our focus while mimicking a real-world AL situation. Future work could explore varying the number of training examples (e.g., 1500, 2000) to understand further the impact of labelled data quantity on active learning strategies in training medical models.

4.1 Baselines

We evaluate our proposed BiCAL against various literature Active Learning strategies:

1. **Random Sampling (RS):** Unlabelled instances are drawn at random.
2. **Normalized Sequence Probability (NSP):** Uses the probability of the generated sequence by the model as a measure of uncertainty.

$$\mathcal{NSP} = 1 - \exp\left\{\frac{1}{n}\sum_{i=1}^{n} log\mathbb{P}(y^i \mid y^1 \ldots, y^n, x)\right\}$$

   (Tsvigun et al., 2023; Wang et al., 2019).

3. **Expected Normalised Sentence Probability (ENSP):** Bayesian AL method where it has the same intuition as NSP.

$$\text{ENSP} = 1 - \mathbb{E}_{w \sim q_{\hat{\theta}}} \bar{p}_w(y|x)$$

   (Tsvigun et al., 2023; Ueffing and Ney, 2007; Wang et al., 2019).

4. **Expected Normalised Sentence Variance (ENSV):** Similar to ENSP but uses variance instead of expectation between the sequence probability.

$$\text{ENSV} = Var_{w \sim q_{\theta}} \bar{p}_w(y|x)$$

   (Tsvigun et al., 2023; Ueffing and Ney, 2007; Wang et al., 2019).

5. **Contrastive Active Learning (CAL):** SOTA AL method described in section 3 (Margatina et al., 2021b).

In addition, for **BiCAL**, we implemented two variants by varying the choice of pre-trained image encoder $\Phi(.)$ in the BiCAL algorithm. We have experimented with two types of pre-trained models, Dinov2 and CheSS, to examine the effect of different types of pre-trained image encoders in our algorithm. Dinov2 is an image model that is pre-trained on a general image dataset (Oquab et al., 2023), whereas CheSS is pre-trained on a CXR dataset (Cho et al., 2023). For the pre-trained text encoder $\Omega(.)$, we have fixed the selection to GatorTron (Yang et al., 2022) based on its SOTA performance in clinical NLP tasks (that outperforms BioBERT (Lee et al., 2019), ClinicalBERT (Huang et al., 2020), BioMegatron (Shin et al., 2020)).

4.2 Datasets

We used the labelled datasets MIMIC-CXR (Johnson et al., 2019a) and IU X-Ray (Demner-Fushman et al., 2015) for our simulation of active learning conditions. The IU X-Ray dataset contains 3,955 radiology reports with 7,470 associated chest X-ray images, while MIMIC-CXR includes 227,835 radiology reports with 377,110 associated chest X-ray images. Following the methodology from Chen et al. (2022), we excluded samples without accompanying reports. We partitioned the IU X-Ray dataset into training and testing sets using an 85%:15% ratio and used the official train-test split for MIMIC-CXR.

In our simulated active learning experiments, we queried only 1,000 data points. As it was impractical in terms of running time to run the experiment on the entire MIMIC-CXR dataset of 377,110 images, we leveraged the structured labels from MIMIC-CXR-JPG (Johnson et al., 2019b) and conducted stratified sampling to obtain a 10% subset of the training split (34,463 data points). This ensured that the subset closely mirrored the label distribution of the full MIMIC-CXR dataset. We used this stratified subset for training and the official test set for evaluation. We release the processed reports with their image IDs for both datasets in CSV files in the repository and provide the data distribution of MIMIC-CXR before and after subset sampling in Table 5 and 6 in the Appendix.

4.3 Setup

Experiments were conducted on a single NVIDIA RTX6000 GPU. We adopted the AdamW optimizer (Loshchilov and Hutter, 2019) with a learning rate of 3e-5 and a weight decay of 3e-7. A warm-up scheduler was applied to the learning rate for the initial 200 steps. Due to computational constraints, we used a training batch size of 8 and limited the maximum number of tokens for generation to 100.

In our experiment, we fine-tuned a vision encoder-decoder model initialized with pre-trained Vision

	Precision	Recall	F-1 Score	Amount of training data
RS	0.450	0.252	0.168	1000
NSP	0.436	0.241	0.194	1000
ENSP	0.558	0.266	0.200	1000
ENSV	0.451	0.268	0.195	1000
CAL	0.326	0.221	0.187	1000
BiCAL Dinov2	0.403	0.255	0.191	1000
BiCAL CheSS	0.429	0.274	0.219	1000
Full Tune	0.309	0.273	0.259	34,463 (full subset)
R2Gen	0.333	0.274	0.276	377,110 (full data)
CCR	0.586	0.237	<0.300*	377,110 (full data)

Table 1: Clinical Efficacy Metrics across AL Strategies after 1000 data queried on MIMIC-CXR Dataset. * stared entries are estimated as the result is not found in the original paper. The best results over AL strategies of each metric are highlighted in blue. The performance of all models is averaged from three runs.

Methods	Example Outputs	BLEU 1-4
Reference	The NG tube extends inferiorly beyond the diaphragm into the fundus of the stomach. Again seen is moderate cardiomegaly. The pulmonary vascular congestion is stable. There are no new focal consolidations. The fissural loculation of pleural fluid along the left chest wall has not changed compared to the prior exam. There is no pneumothorax.	–
RS	The lungs are well expanded and clear. There is no pleural effusion or pneumothorax. The cardiomediastinal silhouette is within normal limits. There is no acute osseous abnormality.	[0.42, 0.36, 0.30, 0.24]
NSP	The heart is normal in size. The mediastinal and hilar contours appear within normal limits. The pulmonary vasculature is normal. There is no pleural effusion or pneumothorax. The lungs appear clear. There is no pulmonary edema.	[0.60, 0.53, 0.43, 0.36]
CAL	The patient is status post median sternotomy, CABG, and mitral valve replacement. The right-sided Port-A-Cath terminates in the low SVC. The right-sided pacemaker terminates in the low SVC. The mediastinal and hilar contours are unchanged. There is no pleural effusion or pneumothorax. The cardiac silhouette is top normal. There is no evidence of pneumothorax.	[0.85, 0.71, 0.56, 0.44]
BiCAL CheSS	The lungs are clear without focal consolidation, effusion, or pneumothorax. The cardiac and mediastinal silhouettes are within normal limits. No acute osseous abnormalities.	[0.37, 0.31, 0.24, 0.19]

Table 2: Case study of Generation Result on Negative Cases using Different AL Methods.

Transformers (ViT) (Dosovitskiy et al., 2020) and GPT-2 (Radford et al., 2019). These models were chosen for their popularity and strong performance in computer vision and natural language processing, respectively. Our primary focus was to investigate active learning strategies in a multi-modal task, so we did not explore other model choices. We utilized HuggingFace (Wolf et al., 2020) and Deepspeed (Rasley et al., 2020) to facilitate our experiment setup.

5 Results and Analysis

We used two types of evaluation metrics: natural language generation (NLG) metrics and domain-specific (clinical efficacy) metrics. This provided a comprehensive evaluation of the generated reports in terms of general and domain-specific performance. For NLG metrics, we reported BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) scores at each active learning iteration. For clinical efficacy metrics, we used the CheXpert (Irvin et al., 2019) model to label the generated and reference reports. We reported precision, recall, and F1 scores for the labeled categories of the generated and reference reports. This evaluation approach is widely used in chest X-ray clinical report generation tasks (Chen et al., 2022; Liu et al., 2019, 2021).

5.1 Clinical Efficacy Metrics

We first assessed the baseline methods and our strategy after 1000 queries on MIMIC-CXR using domain-specific metrics to examine the performance of active learning (AL) strategies, which is crucial for training clinical models. Table 1 displays the clinical efficacy metrics of various AL strategies based on 1000 data queries from a MIMIC-CXR dataset subset. The table's last three rows show the performance of our underlying model after fine-tuning for 10 epochs on the full MIMIC-CXR dataset subset, R2Gen (Chen et al., 2022), and the model (CCR) from Liu et al. (2019). These latter two are fully supervised models trained on the full MIMIC-CXR dataset, designed to excel in chest radiology report generation tasks, with their performance referenced directly from their published papers.

A notable observation is that BiCAL CheSS surpassed baseline methods in recall and F1 scores while maintaining a competitive average precision score. This suggests that the BiCAL CheSS approach effectively recognizes more rare cases (un-

Method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L
CAL	0.4978	0.4177	0.3313	0.2685	0.3115	0.0996	0.2143
RS	0.4487	0.3762	0.3008	0.2456	0.3040	0.0979	0.2138
NSP	0.4832	0.3997	0.3160	0.2563	0.2994	0.1026	0.2178
ENSP	0.4238	0.3569	0.2868	0.2355	0.3066	0.1013	0.2205
ENSV	0.3588	0.3060	0.2477	0.2047	0.2939	0.0969	0.2119
BiCAL Dinov2	0.5025	0.4200	0.3343	0.2726	0.3096	0.1001	0.2183
BiCAL CheSS	0.3930	0.3299	0.2636	0.2153	0.2870	0.0905	0.2078

Table 3: Average NLG performance of different AL strategies after 1000 queries on MIMIC-CXR

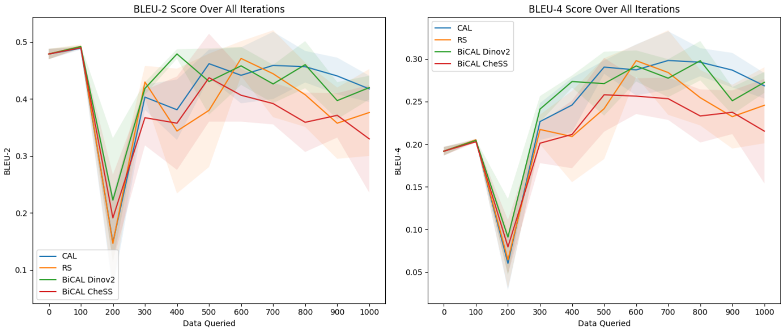


Figure 2: Average NLG Performance of AL Strategies and Best-performing Baselines on MIMIC-CXR

healthy scenarios) than other AL strategies, though it may occasionally increase false positives, as indicated by the precision score. In medical diagnostics, catching every potential disease case (reducing false negatives) is crucial. Therefore, high recall is preferable to high precision, making BiCAL's performance desirable in our context and demonstrating BiCAL CheSS's superiority in generating clinically accurate reports.

Remarkably, the BiCAL CheSS method achieved a recall score that surpasses models fine-tuned on the entire MIMIC-CXR subset (Full Tune). Additionally, it achieved competitive performance with fully supervised models R2Gen and CCR, with a better recall score and an F1 score not much lower. This is noteworthy, considering this performance was achieved with only 1000 data points (less than 0.3% of the whole MIMIC-CXR).

An interesting observation is that although CAL performed well in the NLG metrics on the MIMIC-CXR dataset (Figure 3), its clinical precision and recall scores were the least impressive among all methods. This suggests that while CAL trains models to produce seemingly accurate reports, these might not be clinically sound. By augmenting the contrastive bidirectionally and utilizing pre-trained encoders, the domain-specific performance of this contrastive active learning approach is largely enhanced, demonstrating the success of our approach. We include a case study of generation performance on rare cases using various AL methods in Table 7 in the Appendix.

Furthermore, evidence of the task's complexity is seen in the last three rows of Table 1. These rows include results from R2Gen and CCR, models specifically tailored for chest X-ray report generation and comprehensively trained on the full MIMIC-CXR dataset. Despite their specialized design, their clinical performance remains relatively low. This underscores the inherent challenge of our downstream task—clinical report generation. The intricacies in medical images may be difficult for the underlying model's capability to learn, suggesting that to truly elevate clinical accuracy, superior clinical models adept at the task may need to be designed.

5.2 Natural Language Generation Metrics

We found that for the IU X-ray dataset, no single strategy consistently outperformed the others. Notably, RS and NSP showed marginally better performance during the initial four iterations in both BLEU and ROUGE metrics. For the MIMIC-CXR dataset, CAL performed slightly better in ROUGE scores, while BiCAL was competitive with CAL in BLEU scores, as shown in Figure 3.

The varying performance of CAL across the MIMIC-CXR and IU X-ray datasets suggests that CAL's superiority did not extend to the IU X-ray dataset. This may be due to the different data volumes. Smaller datasets result in a limited unlabeled data pool, potentially narrowing batch sample variance and minimizing observable performance variance. Consequently, the queried batches of different AL strategies on the IU X-ray dataset have more overlap than on MIMIC-CXR, leading to similar performance across strategies.

For the BLEU score, BiCAL Dinov2 performed better than all strategies before 500 queries but was surpassed by CAL afterwards (≥ 500), though it remained competitive. For ROUGE scores, CAL consistently retained slightly better performance starting from 300 queried data. This comparison demonstrates BiCAL's competitiveness in NLG metrics. As shown in Table 3, after 1000 queries, BiCAL Dinov2 achieved the best performance in all BLEU scores and the second-best performance in all ROUGE scores.

Although BiCAL only surpassed other literature AL methods in some NLG metrics, it remained competitive with the best-performing baseline methods. However, language models have been criticized for producing hallucinated text (Ouyang et al., 2022; Stiennon et al., 2020; Ziegler et al., 2019). In a medical setting, our priority is creating accurate clinical reports, not just authoritative-sounding ones. We believe the relatively worse performance of BiCAL CheSS is due to the hallucination problem of LLMs.

CAL and other methods suffer from class imbalance data and may select more healthy cases for training, leading to hallucinated models, that are good at generating good negative (healthy) reports containing many common phrases. In contrast, BiCAL may have a higher proportion of positive cases, training a model with higher clinical efficacy. However, this model's ability to write comprehensive healthy reports that match the reference deteriorates. This results in worse performance on NLG metrics due to the class imbalance problem (more negative cases than positive in the test set, causing the model to generate negative reports more often). This hypothesis is supported by our analysis of the generation results of the models under different active learning methods, including a case study in Table 2. It can be seen that although all reports are saying the candidate contains no significant diseases, but other methods learn to give a more comprehensive healthy report, which results in a higher BLEU score. Thus due to the imbalanced dataset problem, the average NLG score of the other methods may exceed BiCAL despite being less clinically accurate in positive cases (shown in clinical efficacy metrics in Table 1). We also include a positive case study from our analysis to show BiCAL's ability to train clinically accurate models in Table 7.

c	Precision	Recall	F-1 Score
0	0.381	0.254	0.177
0.25	0.376	0.241	0.170
0.50	0.430	0.274	0.219
0.75	0.516	0.250	0.188
1	0.417	0.264	0.199

Table 4: Micro Average of Precision, Recall, and F-1 Score on CheXpert classification Result of BiCAL using different contrastive ratio c after 1000 data queried on MIMIC-CXR Dataset

5.3 Ablation Study

In Sections 5.1 and 5.2, we discussed the impact of different image encoders on the BiCAL algorithm, comparing those pre-trained on a general image dataset (Dinov2) and a Chest X-ray dataset (CheSS). Additionally, a crucial component of the BiCAL algorithm is the contrastive ratio, denoted as c, which determines the sampling ratio between two contrastive definitions in a batch. Our previous experiments used a default c value of 0.5, meaning an equal split between the two contrastive definitions. As shown in Table 4, for clinical efficacy metrics, BiCAL performs best when c is 0.5 in terms of clinical recall and F1 scores. For clinical precision, a c value of 0.75 seems optimal. The poorest performance in terms of clinical recall is observed at c = 0.25. This suggests that while a c value of 0.5 may not be the best for NLG metrics, it ensures the generation of higher clinical quality reports by achieving the best recall of diseases in the generated reports.

6 Conclusion

In this work, we present a study on the effectiveness of current active learning methods for domain-specific multi-modal learning, specifically on the task of clinical report generation from chest X-ray images. We identified the challenge of class imbalance in domain-specific active learning and addressed it by introducing BiCAL, a new active learning technique. BiCAL excelled in both NLG and domain-specific (clinical efficacy) metrics, notably outperforming baselines in clinical recall and F1-score.

We found that existing AL strategies demonstrate similar performance in NLG metrics for the task of clinical report generation from chest X-ray images. This may be due to the complexity of our task, which requires training the model to acquire clinical expertise to generate accurate and clinically sound reports. Interestingly, our tests revealed that an AL strategy's high performance in NLG metrics does not ensure equal success in domain-specific (clinical) performance, possibly due to the hallucination properties of language models. We hope this work provides valuable insights and can act as a starting point for researchers in the future on the task of active learning in multi-modal clinical tasks.

Ethical Consideration and Limitations

We note that despite the success of BiCAL in our study of clinical report generation, in practice, its performance is yet to be confirmed. We have simulated our experiments based on a labelled dataset where the radiology report was collected under a monitored condition such that their format may achieve a certain level of consistency (Johnson et al., 2019a; Demner-Fushman et al., 2015). However, in practice, the queried data's label report may vary based on different radiologist labellers, which may cause noise in the training dataset, which may affect the effectiveness of BiCAL.

We identify that for this work have used sensitive personal data that is related to the health sector. We used MIMIC-CXR (Johnson et al., 2019a) and IU X-Ray (Demner-Fushman et al., 2015) datasets in this project. We note that both datasets have been de-identified, where they have removed all personal health information (PHI). This has ensured the privacy and confidentiality of the individuals. During this project, we handled the data responsibility and used it only for the purpose of research. No attempt at re-identification of the datasets is made. We have also signed the data use agreement for MIMIC-CXR before we use the data. We note that MIMIC-CXR and IU X-rays, just like all datasets, may contain inherent biases based on patient information such as where the data is collected. Moreover, active learning is a technique that samples data based on a certain heuristic, which therefore may introduce additional bias in the sampling and training of the model. This work researches the effectiveness of active learning in clinical report generation, we recognize this potential bias that may be introduced by our research, and this also comes along with our work's contribution to the improvement of the field of active learning in the clinical sector.

Due to the difficulties in acquiring publicly available domain-specific image-report pair data, we chose to work with the task of clinical report generation from chest X-rays. As we designed the BiCAL algorithm, it was not tailored to the clinical report generation task that we conducted our experiments. Moreover, we believe that with the intricacies and high level of expertise required in the medical domain, we believe experiments conducted in this domain can provide valuable insight and act as a good reference for AL's performance on domain-specific learning in general. However, further work should be done in the future to test the performance of BiCAL in other domains, given that the data is available.

References

Jordan T. Ash and Ryan P. Adams. 2020. On warm-starting neural network training. *Preprint*, arXiv:1910.08475.

Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. 2019. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*.

Yutong Bai, Xinyang Geng, Karttikeya Mangalam, Amir Bar, Alan Yuille, Trevor Darrell, Jitendra Malik, and Alexei A Efros. 2023. Sequential modeling enables scalable learning for large vision models. *Preprint*, arXiv:2312.00785.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. 2015. Weight uncertainty in neural networks. *Preprint*, arXiv:1505.05424.

Zhihong Chen, Yan Song, Tsung-Hui Chang, and Xiang Wan. 2022. Generating radiology reports via memory-driven transformer. *Preprint*, arXiv:2010.16056.

Kyungjin Cho, Ki Duk Kim, Yujin Nam, Jiheon Jeong, Jeeyoung Kim, Changyong Choi, Soyoung Lee, Jun Soo Lee, Seoyeon Woo, Gil-Sun Hong, Joon Beom Seo, and Namkug Kim. 2023. CheSS: Chest x-ray pre-trained model via self-supervised contrastive learning. *Journal of Digital Imaging*.

Gui Citovsky, Giulia DeSalvo, Claudio Gentile, Lazaros Karydas, Anand Rajagopalan, Afshin Rostamizadeh, and Sanjiv Kumar. 2021. Batch active learning at scale. *Preprint*, arXiv:2107.14263.

Thomas Cover and Peter Hart. 1967. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27.

Dina Demner-Fushman, Marc Kohli, Marc Rosenman, Sonya Shooshan, Laritza Rodriguez, Sameer Antani, George Thoma, and Clement Mcdonald. 2015. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association : JAMIA*, 23.

Liat Ein Dor, Alon Halfon, Ariel Gera, Eyal Shnarch, Lena Dankin, Leshem Choshen, Marina Danilevsky, Ranit Aharonov, Yoav Katz, and Noam Slonim. 2020. Active learning for bert: An empirical study. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7949–7962.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Meng Fang, Yuan Li, and Trevor Cohn. 2017. Learning how to active learn: A deep reinforcement learning approach. *Preprint*, arXiv:1708.02383.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. 2017. Deep bayesian active learning with image data. In *International conference on machine learning*, pages 1183–1192. PMLR.

Alexios Gidiotis and Grigorios Tsoumakas. 2021a. Bayesian active summarization. *Preprint*, arXiv:2110.04480.

Alexios Gidiotis and Grigorios Tsoumakas. 2021b. Uncertainty-aware abstractive summarization. *ArXiv*, abs/2105.10155.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. 2011. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2020. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *Preprint*, arXiv:1904.05342.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, Jayne Seekins, David A. Mong, Safwan S. Halabi, Jesse K. Sandberg, Ricky Jones, David B. Larson, Curtis P. Langlotz, Bhavik N. Patel, Matthew P. Lungren, and Andrew Y. Ng. 2019. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. *Preprint*, arXiv:1901.07031.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019a. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317.

Alistair EW Johnson, Tom J Pollard, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Yifan Peng, Zhiyong Lu, Roger G Mark, Seth J Berkowitz, and Steven Horng. 2019b. Mimic-cxr-jpg, a large publicly available database of labeled chest radiographs. *arXiv preprint arXiv:1901.07042*.

Seokhwan Kim, Yu Song, Kyungduk Kim, Jeong-Won Cha, and Gary Geunbae Lee. 2006. MMR-based active machine learning for bio named entity recognition. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 69–72, New York City, USA. Association for Computational Linguistics.

Solomon Kullback and R. A. Leibler. 1951. On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

David D Lewis. 1995. A sequential algorithm for training text classifiers: Corrigendum and additional data. In *Acm Sigir Forum*, volume 29, pages 13–19. ACM New York, NY, USA.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Fenglin Liu, Changchang Yin, Xian Wu, Shen Ge, Ping Zhang, and Xu Sun. 2021. Contrastive attention for automatic chest x-ray report generation. In *Findings*.

Guanxiong Liu, Tzu-Ming Harry Hsu, Matthew McDermott, Willie Boag, Wei-Hung Weng, Peter Szolovits, and Marzyeh Ghassemi. 2019. Clinically accurate chest x-ray report generation. *Preprint*, arXiv:1904.02633.

Ming Liu, Wray L. Buntine, and Gholamreza Haffari. 2018. Learning how to actively learn: A deep imitation learning approach. In *Annual Meeting of the Association for Computational Linguistics*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *Preprint*, arXiv:1711.05101.

Katerina Margatina, Loïc Barrault, and Nikolaos Aletras. 2021a. On the importance of effectively adapting pretrained language models for active learning. In *Annual Meeting of the Association for Computational Linguistics*.

Katerina Margatina, Giorgos Vernikos, Loïc Barrault, and Nikolaos Aletras. 2021b. Active learning by acquiring contrastive examples. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 650–663, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. 2023. Dinov2: Learning robust visual features without supervision. *Preprint*, arXiv:2304.07193.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Yotam Perlitz, Ariel Gera, Michal Shmueli-Scheuer, Dafna Sheinwald, Noam Slonim, and Liat Ein-Dor. 2023. Active learning for natural language generation. *Preprint*, arXiv:2305.15040.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. KDD '20, page 3505–3506, New York, NY, USA. Association for Computing Machinery.

Ozan Sener and Silvio Savarese. 2018. Active learning for convolutional neural networks: A core-set approach. *Preprint*, arXiv:1708.00489.

Burr Settles. 2009. Active learning literature survey.

Claude Elwood Shannon. 2001. A mathematical theory of communication. *ACM SIGMOBILE mobile computing and communications review*, 5(1):3–55.

Artem Shelmanov, Dmitri Puzyrev, Lyubov Kupriyanova, Denis Belyakov, Daniil Larionov, Nikita Khromov, Olga Kozlova, Ekaterina Artemova, Dmitry V. Dylov, and Alexander Panchenko. 2021. Active learning for sequence tagging with deep pre-trained models and bayesian uncertainty estimates. *Preprint*, arXiv:2101.08133.

Yanyao Shen, Hyokun Yun, Zachary C Lipton, Yakov Kronrod, and Animashree Anandkumar. 2017. Deep active learning for named entity recognition. *arXiv preprint arXiv:1707.05928*.

Hoo-Chang Shin, Yang Zhang, Evelina Bakhturina, Raul Puri, Mostofa Patwary, Mohammad Shoeybi, and Raghav Mani. 2020. Biomegatron: Larger biomedical domain language model. *Preprint*, arXiv:2010.06060.

Emma Slade and Kim M. Branson. 2022. Deep reinforced active learning for multi-class image classification. *Preprint*, arXiv:2206.13391.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021.

Rinyoichi Takezoe, Xu Liu, Shunan Mao, Marco Tianyu Chen, Zhanpeng Feng, Shiliang Zhang, and Xiaoyu Wang. 2023. Deep active learning for computer vision: Past and future. *APSIPA Transactions on Signal and Information Processing*, 12(1).

Akim Tsvigun, Ivan Lysenko, Danila Sedashov, Ivan Lazichny, Eldar Damirov, Vladimir Karlov, Artemy Belousov, Leonid Sanochkin, Maxim Panov, Alexander Panchenko, Mikhail Burtsev, and Artem Shelmanov. 2023. Active learning for abstractive text summarization. *Preprint*, arXiv:2301.03252.

Nicola Ueffing and Hermann Ney. 2007. Word-level confidence estimation for machine translation. *Computational Linguistics*, 33(1):9–40.

Thuy-Trang Vu, Ming Liu, Dinh Q. Phung, and Gholamreza Haffari. 2019. Learning how to active learn by dreaming. In *Annual Meeting of the Association for Computational Linguistics*.

Shuo Wang, Yang Liu, Chao Wang, Huanbo Luan, and Maosong Sun. 2019. Improving back-translation with uncertainty-based confidence estimation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 791–802, Hong Kong, China. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing. *Preprint*, arXiv:1910.03771.

Xi Yang, Aokun Chen, Nima PourNejatian, Hoo Chang Shin, Kaleb E Smith, Christopher Parisien, Colin Compas, Cheryl Martin, Mona G Flores, Ying Zhang, Tanja Magoc, Christopher A Harle, Gloria Lipori, Duane A Mitchell, William R Hogan, Elizabeth A Shenkman, Jiang Bian, and Yonghui Wu. 2022. Gatortron: A large clinical language model to unlock patient information from unstructured electronic health records. *Preprint*, arXiv:2203.03540.

Michelle Yuan, Hsuan-Tien Lin, and Jordan Boyd-Graber. 2020. Cold-start active learning through self-supervised language modeling. *arXiv preprint arXiv:2010.09535*.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

7 Appendix

Table 5: Label Distribution for Full MIMIC-CXR Dataset

	-1.0	0.0	1.0	N/A
Atelectasis	4.53%	0.67%	20.11%	74.69%
Cardiomegaly	2.65%	6.98%	19.68%	70.68%
Consolidation	1.90%	3.50%	4.73%	89.87%
Edema	5.78%	11.25%	11.86%	71.10%
Enlarged Cardiomediastinum	4.11%	2.32%	3.15%	90.42%
Fracture	0.24%	0.39%	1.93%	97.44%
Lung Lesion	0.50%	0.38%	2.76%	96.36%
Lung Opacity	1.68%	1.35%	22.62%	74.36%
No Finding	0.00%	0.00%	33.12%	66.88%
Pleural Effusion	2.55%	11.92%	23.83%	61.69%
Pleural Other	0.34%	0.06%	0.88%	98.73%
Pneumonia	8.03%	10.68%	7.27%	74.02%
Pneumothorax	0.50%	18.59%	4.55%	76.36%
Support Devices	0.10%	1.53%	29.21%	69.15%

Table 6: Label Distribution for Stratified Subset of MIMIC-CXR Dataset

	-1.0	0.0	1.0	N/A
Atelectasis	4.62%	0.72%	19.94%	74.72%
Cardiomegaly	2.62%	6.83%	19.82%	70.73%
Consolidation	1.83%	3.52%	4.62%	90.03%
Edema	5.79%	11.53%	11.51%	71.17%
Enlarged Cardiomediastinum	4.06%	2.29%	3.10%	90.55%
Fracture	0.24%	0.38%	1.93%	97.45%
Lung Lesion	0.55%	0.42%	2.64%	96.38%
Lung Opacity	1.68%	1.40%	22.71%	74.21%
No Finding	0.00%	0.00%	33.26%	66.74%
Pleural Effusion	2.57%	11.99%	23.54%	61.90%
Pleural Other	0.32%	0.06%	0.87%	98.75%
Pneumonia	8.09%	10.56%	7.39%	73.97%
Pneumothorax	0.50%	18.36%	4.65%	76.48%
Support Devices	0.09%	1.48%	29.43%	69.00%

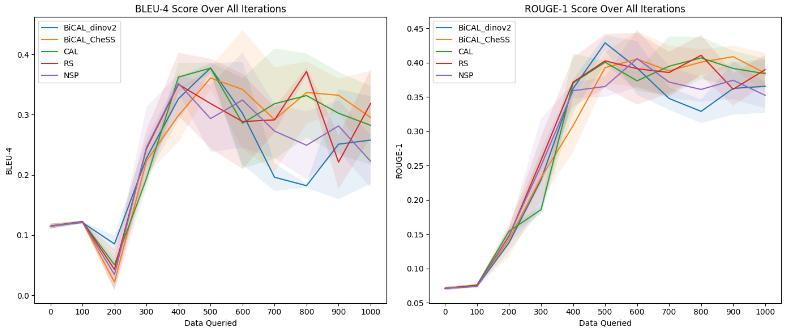


Figure 3: Average NLG Performance of AL Strategies and Best-performing Baselines on IU X-ray

Methods	Example Outputs
Reference	Lung volumes are low. Mild to moderate enlargement cardiac silhouette is unchanged, accentuated by the presence of low lung volumes. The aorta remains tortuous. Mediastinal and hilar contours are stable. There is continued mild pulmonary vascular congestion without overt pulmonary edema. Patchy and linear opacities in the lung bases likely reflect areas of atelectasis. No pneumothorax or pleural effusion is clearly evident. Percutaneous gastrostomy catheter is incompletely imaged.
RS	The lungs are clear. There is no pleural effusion or pneumothorax. Cardiomediastinal silhouette is within normal limits. No acute osseous abnormalities.
NSP	The heart is normal in size. The mediastinal and hilar contours appear within normal limits. There is no pneumothorax. The pulmonary vasculature is normal. There is no pleural effusion or pneumothorax. There is no pneumomediastinum.
CAL	The heart is mildly enlarged. There is mild prominence of pulmonary vascularity with mild interstitial edema. There is no pleural effusion or pneumothorax. The mediastinal and hilar contours are unremarkable. There is no evidence of pneumomediastinum.
BiCAL CheSS	The cardiac silhouette is mildly enlarged. The aorta is tortuous. There is mild cardiomegaly. There is no pleural effusion or pneumothorax. The hilar contours are normal. There is mild pulmonary vascular congestion.

Table 7: Case study of Generation Result on Positive Cases using Different AL Methods. Green: The generated diagnosis is matched with reference. Red: The generated diagnosis is incorrect compared to the reference. Yellow: The generated diagnosis is not mentioned in the reference.

Disease	**RS**	**NSP**	**ENSP**	**ENSV**	**CAL**	**BiCAL Dinov2**	**BiCAL CheSS**	**Full Tune**
No Finding	0.0738	0.0799	0.0766	0.0843	0.0911	0.0750	0.1068	0.1507
Enlarged Cardiomediastinum	0.2183	0.2410	0.2378	0.2333	0.2462	0.2318	0.2386	0.2958
Cardiomegaly	0.2475	0.2592	0.1783	0.2354	0.2781	0.1829	0.4177	0.5113
Lung Lesion	1.0000	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	0.0000
Lung Opacity	1.0000	0.6667	0.6667	1.0000	0.4333	0.3333	0.5000	0.3798
Edema	0.1781	0.1869	0.1555	0.1548	0.1757	0.1669	0.1584	0.2315
Consolidation	0.2879	0.4248	0.3455	0.3292	0.3029	0.3241	0.2981	0.3160
Pneumonia	0.2000	0.1221	1.0000	0.1176	0.0870	0.0000	0.1481	0.0887
Atelectasis	0.3846	0.3509	0.3636	0.3333	0.2773	0.5000	0.3333	0.2739
Pneumothorax	0.5621	0.6102	0.5876	0.5701	0.5569	0.5713	0.5917	0.5949
Pleural Effusion	0.4567	0.5131	0.4949	0.4945	0.4558	0.4906	0.4876	0.6016
Pleural Other	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.0000
Fracture	0.0000	0.0000	1.0000	1.0000	0.0323	0.0000	0.0000	0.1667
Support Devices	0.6939	0.6418	0.6986	0.7545	0.6253	0.7610	0.7282	0.7096
Macro Average	0.4502	0.4355	0.5575	0.4505	0.3258	0.4026	0.4292	0.3086

Table 8: Precision on CheXpert classification Result between reference and generated report across AL Strategies after 1000 queries on MIMIC-CXR

Disease	**RS**	**NSP**	**ENSP**	**ENSV**	**CAL**	**BiCAL Dinov2**	**BiCAL CheSS**	**Full Tune**
No Finding	0.9042	0.8314	0.9042	0.8391	0.7011	0.7893	0.6590	0.7356
Enlarged Cardiomediastinum	0.4196	0.3924	0.4030	0.3970	0.3587	0.4267	0.4237	0.3869
Cardiomegaly	0.1221	0.1512	0.1042	0.1753	0.2267	0.1945	0.4083	0.3757
Lung Lesion	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Lung Opacity	0.0005	0.0010	0.0010	0.0010	0.0199	0.0005	0.0005	0.1921
Edema	0.1357	0.1614	0.1801	0.1376	0.0752	0.1402	0.1961	0.1145
Consolidation	0.6344	0.1336	0.4760	0.5180	0.2395	0.4812	0.5120	0.2372
Pneumonia	0.0022	0.0229	0.0000	0.0022	0.0131	0.0000	0.0218	0.0196
Atelectasis	0.0041	0.0164	0.0296	0.0008	0.0961	0.0041	0.0008	0.0895
Pneumothorax	0.7024	0.8285	0.7880	0.8968	0.7810	0.7900	0.8297	0.5608
Pleural Effusion	0.5581	0.5395	0.5318	0.6064	0.4310	0.5322	0.5302	0.6205
Pleural Other	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Fracture	0.0000	0.0000	0.0000	0.0000	0.0034	0.0000	0.0000	0.0034
Support Devices	0.0400	0.2928	0.3039	0.1717	0.1452	0.2040	0.2551	0.4797
Macro Average	0.2517	0.2408	0.2658	0.2676	0.2208	0.2545	0.2741	0.2725

Table 9: Recall on CheXpert classification Result between reference and generated report across AL Strategies after 1000 queries on MIMIC-CXR

Disease	**RS**	**NSP**	**ENSP**	**ENSV**	**CAL**	**BiCAL Dinov2**	**BiCAL CheSS**	**Full Tune**
No Finding	0.1365	0.1458	0.1412	0.1531	0.1612	0.1370	0.1838	0.2502
Enlarged Cardiomediastinum	0.2872	0.2986	0.2991	0.2939	0.2920	0.3004	0.3053	0.3353
Cardiomegaly	0.1635	0.1910	0.1315	0.2010	0.2498	0.1886	0.4129	0.4331
Lung Lesion	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Lung Opacity	0.0010	0.0020	0.0020	0.0020	0.0381	0.0010	0.0010	0.2552
Edema	0.1540	0.1732	0.1669	0.1457	0.1054	0.1524	0.1753	0.1532
Consolidation	0.3961	0.2033	0.4004	0.4026	0.2675	0.3873	0.3768	0.2710
Pneumonia	0.0043	0.0385	0.0000	0.0043	0.0227	0.0000	0.0380	0.0321
Atelectasis	0.0081	0.0314	0.0547	0.0016	0.1427	0.0081	0.0016	0.1349
Pneumothorax	0.6245	0.7028	0.6732	0.6971	0.6502	0.6631	0.6907	0.5773
Pleural Effusion	0.5023	0.5260	0.5127	0.5448	0.4431	0.5105	0.5080	0.6109
Pleural Other	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Fracture	0.0000	0.0000	0.0000	0.0000	0.0062	0.0000	0.0000	0.0067
Support Devices	0.0756	0.4021	0.4236	0.2797	0.2357	0.3217	0.3779	0.5724
Macro Average	0.1681	0.1939	0.2004	0.1947	0.1868	0.1907	0.2194	0.2594

Table 10: F1 Score on CheXpert classification Result between reference and generated report across AL Strategies after 1000 queries on MIMIC-CXR

Generation and De-Identification of Indian Clinical Discharge Summaries using LLMs

Sanjeet Singh[‡*] **Shreya Gupta**[†*] **Niralee Gupta**[†]
Naimish Sharma[†] **Lokesh Srivastava**[†] **Vibhu Agarwal**[†]
Ashutosh Modi[‡]
[‡]Indian Institute of Technology Kanpur (IIT Kanpur) [†]Miimansa
{sanjeet, ashutoshm}@cse.iitk.ac.in
{shreya.gupta,niralee.gupta,naimish.sharma}@miimansa.com
{lokesh.srivastava,vibhu}@miimansa.com

Abstract

The consequences of a healthcare data breach can be devastating for the patients, providers, and payers. The average financial impact of a data breach in recent months has been estimated to be close to USD 10 million. This is especially significant for healthcare organizations in India that are managing rapid digitization while still establishing data governance procedures that align with the letter and spirit of the law. Computer-based systems for de-identification of personal information are vulnerable to data drift, often rendering them ineffective in cross-institution settings. Therefore, a rigorous assessment of existing de-identification against local health datasets is imperative to support the safe adoption of digital health initiatives in India. Using a small set of de-identified patient discharge summaries provided by an Indian healthcare institution, in this paper, we report the nominal performance of de-identification algorithms (based on language models) trained on publicly available non-Indian datasets, pointing towards a lack of cross-institutional generalization. Similarly, experimentation with off-the-shelf de-identification systems reveals potential risks associated with the approach. To overcome data scarcity, we explore generating synthetic clinical reports (using publicly available and Indian summaries) by performing in-context learning over Large Language Models (LLMs). Our experiments demonstrate the use of generated reports as an effective strategy for creating high-performing de-identification systems with good generalization capabilities.

1 Introduction

Over 330 million patient records in India have already been linked with a unique central ID (PIB Press Release). To put this in perspective, the number roughly equals the total population of the United States. Several federal initiatives aimed at establishing standards for medical information exchange, adoption of controlled terminologies, and promoting open architecture-based systems for the management of patient records have seen a steady rise in the adoption of electronic health records within Indian healthcare institutions (Ministry of Health and Family Welfare (MoHFW), India; Srivastava, 2016). This data represents an under-utilized resource that has profound implications for informing public policy, medical research and patient care. At the same time, it also poses some serious challenges. The risks of revealing patient identity even from data that has been anonymized are well known (Sweeney, 2013). Privacy regulations such as GDPR 2016 (European Parliament and Council of the European Union) and the HIPAA Privacy Rule 2003 (U.S. Department of Health and Human Services (HHS)) lay down heavy penalties on non compliance with data safety protocols. A robust data de-identification pipeline is vital if we aim to unlock insights from these electronic patient histories.

Natural Language Processing (NLP) methods for de-identification are known to perform significantly better than manual de-identification (Douglass et al., 2004). However, these have been studied mostly in the single-institution setting. There are limited studies that evaluate de-identification performance of these methods across institutions (Yang et al., 2019). These suggest that NLP methods for de-identification perform poorly when evaluated on data from a different institution compared to the one that contributed the training data. This is especially significant in the context of patient data originating within Indian healthcare institutions. To the best of our knowledge, studies evaluating the performance of NLP based de-identification systems on patient data from Indian healthcare institutions have not yet been carried out. One reason for this might be that until recently there was no regulatory framework for accessing patient data for

[*] Equal Contribution

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 342–362
August 16, 2024. ©2024 Association for Computational Linguistics

research. The Indian Digital Personal Data Protection Act 2023 (DPDPA) (Ministry of Electronics and Information Technology (MeitY), India) is a landmark legislation that came into effect in September 2023 and covers all organizations that process the personal data of individuals in India. Similar to GDPR 2016, the DPDPA defines responsibilities for organizations that collect, store, and process data from patients in India and holds them legally accountable for safeguarding patient privacy. The DPDPA also highlights the need for a data de-identification solution that has been validated on patient data from Indian healthcare institutions.

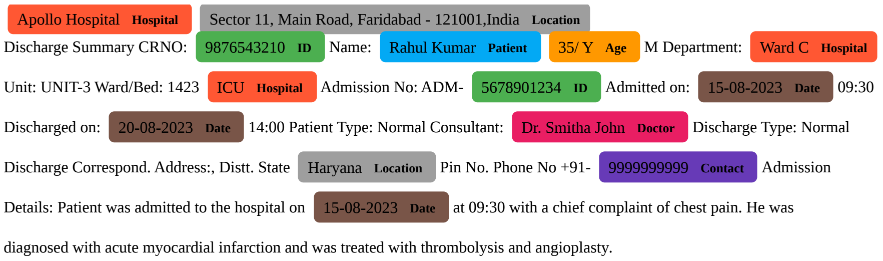
The present study takes a step towards answering this imminent need. Using a dataset of fully de-identified 99 discharge summaries obtained under Institutional Review Board (IRB) approval from the Sanjay Gandhi Post Graduate Institute of Medical Sciences (SGPGIMS), Lucknow, India, the study evaluates language models (LMs) for the task of de-identification. Furthermore, commercially available de-identification solutions are also evaluated. Hereafter, we refer to this dataset as the Indian Clinical Discharge Summaries (ICDS$_R$, subscript R refers to real) dataset. Given the paucity of clinical data, the study also evaluates Large Language Models (LLMs) on the task of generating synthetic clinical texts for training de-identification models. Critically, the study highlights the existence of several personal health information (PHI) elements in the ICDS$_R$ dataset that are unique to the language use and cultural practices in India. It is unlikely that the existing de-identification solutions have been trained to recognize these unique PHI elements, and therefore, their detection may be unreliable. In a nutshell, we make the following contributions:

- We introduce a new dataset (Indian Clinical Discharge Summaries (ICDS$_R$)) obtained from an Indian hospital and evaluate the performance of PI-RoBERTa model (PI-RoBERTa) (fine-tuned on non-Indian clinical summaries) on ICDS$_R$ for the task of De-Identification. Our experiments show poor cross-institutional performance. Experiments with existing commercial off-the-shelf clinical de-identification systems show similar trends.
- To overcome the paucity of Indian clinical data, we generate synthetic summaries using LLMs (Gemini (Team et al., 2023), Gemma (Team et al., 2024), Mistral (Jiang et al., 2023), and Llama3 (Touvron et al., 2023)) via In-Context Learning (ICL). Further, the synthetic summaries are used to train PI-RoBERTa for de-identification on ICDS$_R$. Results show significant improvement in the performance of the de-identification system.
- We release the model code and experiments via GitHub: `https://github.com/Exploration-Lab/llm-for-clinical-report-generation-deidentification`

2 Related Work

Automatic data de-identification methods for biomedical texts have focused on leveraging machine learning techniques to ensure privacy while maintaining data utility. Named Entity Recognition (NER) systems have been tailored to identify and anonymize personal health information/personal identifiable information (PHI/PII) from clinical narratives. Earlier work explored Support Vector Machines (SVMs) for identifying PHI (Neamatullah et al., 2008). Researchers have also explored deep learning models, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) (Dernoncourt et al., 2017), which have shown superior performance over the conventional approach.

In recent years, there has been a growing interest in the application of transformer-based models like BERT (Devlin et al., 2018) for the clinical NER and de-identification task (Chaudhry et al., 2022; Alsentzer et al., 2019). LLMs have also been explored for various clinical tasks such as clinical NLI (Mandal and Modi, 2024). Hybrid approaches that combine rule-based and machine learning methods have also been developed to enhance the robustness of de-identification systems (Meystre et al., 2010). A study by Yang et al. (2019) used a hybrid model combining Long Short-Term Memory (LSTM) networks with Conditional Random Fields (CRFs) for the de-identification of clinical notes. It demonstrated the effectiveness of integrating local resources and diverse word embeddings, and achieved high F1 scores across various de-identification tasks. Furthermore, El Azzouzi et al. (2023) de-identified French electronic health records using distant supervision and deep learning techniques. The study utilized models like Bi-LSTM+CRF and enhanced them with contextualized word embeddings. It achieved remarkable accuracy in removing identifiable information while maintaining data utility. These innovations underscore the continuous improvement



Apollo Hospital Hospital Sector 11, Main Road, Faridabad - 121001,India Location
Discharge Summary CRNO: 9876543210 ID Name: Rahul Kumar Patient 35/ Y Age M Department: Ward C Hospital
Unit: UNIT-3 Ward/Bed: 1423 ICU Hospital Admission No: ADM- 5678901234 ID Admitted on: 15-08-2023 Date 09:30
Discharged on: 20-08-2023 Date 14:00 Patient Type: Normal Consultant: Dr. Smitha John Doctor Discharge Type: Normal
Discharge Correspond. Address:, Distt. State Haryana Location Pin No. Phone No +91- 9999999999 Contact Admission
Details: Patient was admitted to the hospital on 15-08-2023 Date at 09:30 with a chief complaint of chest pain. He was
diagnosed with acute myocardial infarction and was treated with thrombolysis and angioplasty.

Figure 1: A sample of annotated text from Discharge Summary

and adaptation of de-identification methods to address the evolving challenges in data privacy. With remarkable progress made in generative AI techniques, researchers have started exploring generating synthetic clinical data. For example, medGAN (Choi et al., 2017) has been proposed to generate high-dimensional discrete variables such as patient records. It shows that it can produce realistic EHR data that preserves the statistical properties of the original dataset. Researchers have also explored differential privacy techniques in conjunction with Generative Adversarial Networks (GANs) to ensure that the synthetic data does not allow re-identification of individuals. There is also ongoing research into hybrid models that combine rule-based and machine learning techniques to generate data that not only looks realistic but also adheres to known clinical correlations and constraints (Isasa et al., 2024; Goncalves et al., 2020). Such approaches ensure that the synthetic data is both safe and scientifically valid for use in biomedical modeling simulations. The trend highlights the potential of synthetic data to address privacy and data availability challenges in biomedical research. In this paper, we explore LLMs for generating synthetic clinical reports that closely resemble reports in ICDS$_R$, thus capturing the underlying data generation processes.

3 Clinical Discharge Summaries Datasets

n2c2: We make use of the 2006 and 2014 n2c2 datasets (Özlem Uzuner et al., 2008; Stubbs et al., 2015). The 2006 challenge involved the development of automated methods to de-identify discharge summaries from patient medical records (Özlem Uzuner et al., 2008). The total number of summaries in the n2c2-2006 dataset are 888, split between training and test sets. The 2014 challenge comprised of two tasks: de-identification and heart disease risk factor identification (Stubbs et al., 2015). For the de-identification task, the dataset included a variety of clinical documents such as progress notes, discharge summaries, and other narrative texts that typically contain detailed patient information.

Indian Clinical Discharge Summaries (ICDS$_R$): We obtained fully de-identified 99 discharge summaries obtained under Institutional Review Board (IRB) approval from the Sanjay Gandhi Post Graduate Institute of Medical Sciences (SGPGIMS), Lucknow, India. All discharge summaries in the Indian Clinical Corpus were manually annotated for de-identified entities by human annotators using Doccanno (Nakayama et al., 2018), a data annotation tool. Each document was annotated by one annotator. The annotators had previous experience in clinical text annotations. Following established practice, we used the BIO scheme (Ramshaw and Marcus, 1999) for annotating named entities. Our PHI labels were defined by augmenting the PHI entities defined in the HIPAA Privacy Rule 2003 along with adaptation to Indian clinical texts. After annotation, we obtained 26 PHI unique entities in the ICDS$_R$ dataset. Subsequently, due to privacy concerns, PHI elements were replaced with fake values through an automatic replacement tool developed using the Python library *Faker* (Faraglia and Other Contributors, 2010) (example in Fig. 1). Repeated occurrences of an entity within a note were tracked for consistent replacements. Moreover, settings such as date/time offsets were parameterized via a configurable file. The tool provides a scalable solution for de-identifying medical datasets while ensuring secure data access. Table 1 provides statistics of the datasets.

4 Generated Discharge Summaries Datasets

Initial experimentation showed over-fitting in models on the ICDS$_R$ data due to its small size ($69, 10, 20$ summaries for train, val, and test sets, respectively). Consequently, we generated synthetic summaries to augment ICDS$_R$ data. Synthetic patient data is being used increasingly for a variety

Statistics	Training dataset					Test set		
	n2c2-2006	n2c2-2014	ICDS$_R$	ICDS^g_G	ICDS^l_G	n2c2-2006	n2c2-2014	ICDS$_R$
# Summaries	668	790	79	1596	1043	220	514	20
# Unique Tokens	29218	55907	13542	56780	25184	15231	41066	6106
Max Length	3023	2984	9494	4256	2590	2687	2474	8511
Min Length	13	74	97	100	109	15	99	270
Avg. Summary Length	581.71	618.86	1005.94	373.80	392.34	748.22	615.19	1343.40
Original Tag Set	9	24	26	34	106	9	21	24
Mapped Tag set	9	9	9	9	9	9	9	9

Table 1: Statistics of various datasets

of in-silico biomedical experiments in addition to training data augmentation (Chen et al., 2021). Using the samples from ICDS$_R$ we generated medical discharge summaries specific to Indian patients using LLMs (Gemma, Llama-3-8B-Instruct, and Mistral-7B-Instruct-v0.1) via In-Context Learning (ICL). We experimented extensively with various prompts and discharge summaries, as explained below. Our choice of LLMs was driven by the feasibility of instantiating these models on-premise. Prompting is a key aspect of using LLMs. As described below, we experimented with various prompt designs.

Discharge Summaries Generation using the n2c2-2006 dataset: Since the n2c2-2006 discharge summaries are publicly accessible, we generated synthetic discharge summaries based on these along with PHI annotations using Gemini-pro-1.0. We arrived at a functional prompt by iteratively tuning and inspecting the synthesized outputs for overall length, presence of key subsections, and correct PHI annotation. While tuning our prompts, we did not check for the medical validity of the discharge summaries (see App. Table 12). The prompt also contained an original n2c2-2006 summary as an exemplar. This way, we generated five patient discharge summaries for each original discharge summary in the n2c2-2006 dataset and a total of 3000 discharge summaries. The generated summaries were manually reviewed, and the ones containing gibberish text and missing or incorrect annotations were filtered out, resulting in 1596 synthetic discharge summaries with PHI annotations. Hereinafter, we refer to this dataset as ICDS^g_G .

Discharge Summaries Generated using the ICDS$_R$ dataset: The ICDS$_R$ dataset is accessible only under the Institutional Review Board's approval, and therefore, LLMs that can be inferred only via public API endpoints cannot be used to process these. Consequently, we generated synthetic discharge summaries for the ICDS$_R$ dataset only with LLMs that could be instantiated within our secure compute infrastructure (Llama-3-8B-Instruct, Gemma and Mistral-7B-Instruct-v0.1, respectively). We evaluated various LLM and prompt combinations to converge on Llama-3-8B-Instruct (see App. Table 12 for the prompt). To evaluate the performance of model-prompt combinations, we calculated two metrics: BERT F1-Score and the average length of summaries (in words). The BERT F1-Score was calculated on a sample of synthetic annotated discharge summaries (target) and the 99 original ICDS$_R$ discharge summaries (see Table 2). The BERT F1-Score of Meta-Llama-3-8B-Instruct and Mistral-7B-Instruct-v0.1 models with prompt B surpass other model-prompt combinations. We selected the Meta-Llama-3-8B-Instruct model for synthetic discharge summary generation and PHI annotation since, in addition to a high BERT F1-score, the generated summaries are, on average, longer. The ICDS$_R$ dataset was split so that 79 summaries were used in the prompt to generate synthetic summaries while the remaining 20 were reserved for the test set. The temperature parameter of Meta-Llama-3-8B-Instruct was set to 0.9. Around 25 summaries were generated for each of the 79 ICDS$_R$ discharge summaries by embedding these one at a time as an exemplar in the prompt. In total, 1831 discharge summaries, which already had PHI annotations, were generated, yielding 1043 generated discharge summaries after manual review and filtration. Hereinafter, we refer to this dataset as ICDS^l_G. Further, we asked two annotators to annotate 50 generated summaries (after removing the PHI tags) with PHI tags. The Cohen's kappa coefficient (Warrens, 2015), the measure of inter-annotator agreement, was 0.921, showing a high agreement.

Evaluation of the Quality of the Generated Summaries: We assessed the face validity of the gen-

Prompt Id	Model Used	BERT F1-Score	Avg. Summary Length (words)
B	meta-llama/Meta-Llama-3-8B-Instruct	0.491	564
B	mistralai/Mistral-7B-Instruct-v0.1	0.493	400
C	meta-llama/Meta-Llama-3-8B-Instruct	0.486	503
C	mistralai/Mistral-7B-Instruct-v0.1	0.468	267
C	google/gemma-1.1-7b-it	0.478	268

Table 2: Comparison of model-prompt combinations

Training Set	Test Set		
	n2c2-2006	n2c2-2014	ICDS $_R$
n2c2-2006	✓	✓	✓
n2c2-2014	✓	✓	✓
n2c2-2006+ n2c2-2014	✓	✓	✓
ICDS ^g_G	✓	✓	✓
ICDS ^l_G	✓	✓	✓
ICDS ^g_G+ ICDS ^l_G	✓	✓	✓
ICDS ^g_G+ ICDS ^l_G+ n2c2-2014	✓	✓	✓
ICDS ^g_G+ ICDS ^l_G+ n2c2-2014	✓	✓	✓

Table 3: Experiments Matrix

erated summaries by asking physicians to review a convenience sample of 30 real and 30 synthetic discharge summaries with the real/synthetic labels suppressed. The 60 discharge summaries were shuffled and uploaded to a secure, online review tool accessible only to the reviewers (physicians). The reviewers were asked to review each summary and then assign a single label (real or synthetic) to each based on their experience. The review results were compiled, and the precision, recall, and F1 scores were computed for each physician along with Cohen's Kappa to assess agreement between the two physicians (details in §7).

As can be observed in Table 1, for the purpose of uniformity and modeling, we mapped PHI entities in each of the dataset to 9 tags (corresponding to 8 unique entities + 1 OTHERS). App. Table 15 provides details of tag mapping where the PHI entities are mapped with to their superset and all non-PHI entities are mapped to OTHERS Tag.

5 De-Identification Task

De-Identification Task: De-Identification is conceptually similar to a Named Entity Recognition task. Both ICDS ^g_G and ICDS ^l_G were pre-processed and converted into BIO format as is customary in Named Entity Recognition development (also see App. Fig. 4). Formally, given some text, $S = (w_1, w_2, w_3, ..., w_n)$ containing n words, de-identification requires labeling each of the word w_i with a tag t_k coming from a NER tagset $t_1, t_2, ..., t_T$. Subsequently, the labeled entities can be redacted or replaced with fake values for privacy protection.

De-Identification Model: We fine-tuned several different NER models, including ghadeermobasher/BCHEM4-Modified-BioBERT-v1 (BioBERT) and Clinical-AI-Apollo/Medical-NER (Clinical AI Apollo). In each case, we used a training partition of the data to train and a validation partition for evaluation. However, the Clinical NER models did not perform well since they are designed to label medical entities such as disease, drugs, procedures, and devices (see App. D). RoBERTa-NER-Personal-Info model (PI-RoBERTa) showed good performance on n2c2-2006 and n2c2-2014 datasets. The architecture for PI-RoBERTa is shown in App. Fig. 13. PI-RoBERTa is a 24-layered transformer model that predicts a label for each token.

Attribute	Dataset	
	Real	Generated
Counts	3158684	5022667
Length (words)	560753	721886
Mean ± SE	4.64 ± 0.004	5.93 ± 0.005
Median	4.0	5.0
Min	1	1
Max	50	89
Jaccard Distance	0.83	
BERTScore (F1)	0.64	
BERTScore (Precision)	0.65	
BERTScore (Recall)	0.63	

Table 4: Comparison of n2c2-2006 and ICDS ^g_G Dataset

Attribute	Dataset	
	Real	Generated
Counts	636805	4789863
Length (words)	102604	508244
Mean ± SE	5.21 ± 0.01	7.77 ± 0.01
Median	4.0	5.0
Min	1	1
Max	72	472
Jaccard Distance	0.80	
BERTScore (F1)	0.58	
BERTScore (Precision)	0.60	
BERTScore (Recall)	0.56	

Table 5: Comparison of ICDS $_R$ and ICDS ^l_G Dataset

6 Model Training Experiments

Initial experiments with ICDS $_R$ using a 69-10-20 (train-val-test) split resulted in overfitting given that ICDS $_R$ is small, containing only 99 discharge summaries. We also experimented with training the model on n2c2-2006 and n2c2-2014 datasets and testing on ICDS $_R$ to check for cross-institutional generalization. We experimented with several combinations of real and synthetic datasets and evaluated on the test set of n2c2-2006, n2c2-2014, and ICDS $_R$. Table 3 shows the experiments matrix, in total we evaluated 24 different combinations. For all the experiments, we reserved 20 summaries of ICDS $_R$ for testing. Note that these summaries were also not used for generation. For each experiment, PI-RoBERTa was fine-tuned on each training set as

Training Data	n2c2-2006			n2c2-2014			n2c2-2006 + n2c2-2014		
Testing Data	n2c2-2006	n2c2-2014	ICDS$_R$	n2c2-2006	n2c2-2014	ICDS$_R$	n2c2-2006	n2c2-2014	ICDS$_R$
CONTACT	0.98	0.66	0.18	0.73	0.95	0.20	0.96	0.93	0.24
PATIENT	0.95	0.65	0.81	0.82	0.98	0.85	0.91	0.96	0.77
DOCTOR	0.95	0.89	0.64	0.93	0.98	0.76	0.97	0.98	0.54
ID	0.99	0.55	0.64	0.96	0.97	0.65	1.00	0.96	0.93
DATE	0.98	0.43	0.16	0.70	0.99	0.97	0.97	0,98	0.97
LOCATION	0.89	0.80	0.71	0.78	0.95	0.80	0.81	0.94	0.75
HOSPITAL	0.94	0.79	0.34	0.87	0.94	0.36	0.94	0.93	0.40
AGE	0.80	0.00	0.00	0.02	0.99	0.48	0.12	0.94	0.53
Micro Avg	0.96	0.66	0.41	0.81	0.98	0.80	0.96	0.97	0.80
Macro Avg	0.93	0.60	0.43	0.72	0.97	0.63	0.83	0.95	0.64
Weighted Avg	0.96	0.61	0.31	0.84	0.98	0.78	0.96	0.97	0.78

Table 6: F1 scores for PHI entities with overall micro Avg F1 , macro Avg F1 , Weighted Avg F1

Training Data	ICDS^g_G			ICDS^l_G			ICDS^g_G+ ICDS^l_G		
Testing Data	n2c2-2006	n2c2-2014	ICDS$_R$	n2c2-2006	n2c2-2014	ICDS$_R$	n2c2-2006	n2c2-2014	ICDS$_R$
CONTACT	0.80	0.47	0.11	0.55	0.38	0.96	0.93	0.67	0.98
PATIENT	0.74	0.56	0.68	0.05	0.32	0.95	0.83	0.60	0.90
DOCTOR	0.86	0.78	0.88	0.35	0.71	0.98	0.86	0.76	0.98
ID	0.87	0.58	0.51	0.81	0.61	1.00	0.93	0.63	0.98
DATE	0.87	0.90	0.88	0.70	0.84	0.99	0.90	0.88	0.99
LOCATION	0.71	0.78	0.34	0.50	0.66	0.97	0.75	0.81	0.96
HOSPITAL	0.87	0.72	0.31	0.42	0.51	0.97	0.88	0.70	0.98
AGE	0.02	0.67	0.51	0.02	0.38	0.96	0.06	0.56	0.97
Micro Avg	0.85	0.77	0.68	0.55	0.67	0.98	0.88	0.76	0.98
Macro Avg	0.72	0.68	0.53	0.42	0.55	0.97	0.77	0.70	0.97
Weighted Avg	0.86	0.77	0.69	0.52	0.66	0.98	0.88	0.77	0.98

Table 7: F1 scores for PHI entities for the PI-RoBERTa trained on generated data.

given in Table 3 and tested on each corresponding test set. Details about training are given in App. D

Comparison with Commercial De-Identification Systems: We compared the performance of these on the ICDS$_R$ test set. In particular, we evaluated AWS's (Amazon Web Services) Comprehend Medical DetectPHI (Amazon Web Services) and GCP's (Google Cloud Platform) Data Loss Protection (DLP) (Google Cloud) de-identification solutions. For comparison and evaluation, ICDS$_R$ test set was mapped to a common tag set, which includes DATE, NAME, LOCATION, AGE, ID, and CONTACT. To ensure consistency across the dataset, pre-processing steps were applied. For instance, titles such as 'Dr.' and 'Mr.' were removed from NAME entities in the ICDS$_R$ test set due to the solution's inability to recognize them. Certain tags and entities were excluded from the analysis to align with a common tag set. The LOCATION entity was standardized by merging all location-related entities (street, city, state, zip) into a single LOCATION entity. Similarly, HOSPITAL, ORGANISATION_NAME and ADDRESS entities were consistently mapped to LOCATION.

De-identification using LLMs: We further evaluated the performance of LLMs on ICDS$_R$ test set. Meta-Llama-3-8B-Instruct was instantiated within our secure compute infrastructure, and the prompt was developed for medical text de-identification using the iterative approaches described in the foregoing sections.

7 Experiments, Results and Analysis

Comparison of datasets: The total number of summaries in the n2c2-2006 dataset are 888, split between training and test sets, as shown in Table 1. The n-gram analysis of the n2c2-2006 and ICDS$_R$ datasets reveals distinct linguistic patterns reflecting their unique clinical foci. The n2c2-2006 dataset features unigrams like 'patient,' 'discharge,' and medication-related terms such as 'mg' and 'po' and bigrams like 'mg po' and 'discharge date,' highlighting a narrative centered on patient management and clinical processes as shown in App. Fig. 14. In contrast, the ICDS$_R$ dataset (as shown in App. Fig. 18) shows a marked presence of terms

Training Data	n2c2-2014+ ICDS^l_G+ ICDS^g_G			n2c2-2014+ n2c2-2006+ ICDS^g_G+ ICDS^l_G		
Testing Data	n2c2-2006	n2c2-2014	ICDS$_R$	n2c2-2006	n2c2-2014	ICDS$_R$
CONTACT	0.89	0.95	0.98	0.97	0.96	0.98
PATIENT	0.87	0.97	0.88	0.94	0.96	0.88
DOCTOR	0.95	0.97	0.98	0.98	0.98	0.99
ID	0.99	0.97	0.99	0.99	0.96	0.99
DATE	0.82	0.99	0.99	0.99	0.99	0.99
LOCATION	0.76	0.95	0.98	0.85	0.94	0.98
HOSPITAL	0.93	0.94	0.96	0.96	0.94	0.97
AGE	0.02	0.97	0.96	0.35	0.97	0.86
Micro Avg	0.88	0.97	0.97	0.97	0.97	0.97
Macro Avg	0.78	0.96	0.96	0.88	0.96	0.96
Weighted Avg	0.90	0.97	0.97	0.97	0.97	0.97

Table 8: F1 scores of PHI entities when PI-RoBERTa is fine-tuned on combination of datasets

Metric	AWS	GCP
F1 Score	0.37	0.47

Table 9: Results: AWS vs. GCP Solutions on ICDS$_R$ test set

Entity	AWS	GCP
DATE	0.39	0.56
NAME	0.57	0.52
LOCATION	0.20	0.22
AGE	0.12	0.00
ID	0.17	0.17
CONTACT	0.63	0.36

Table 10: F1 scores for Entity-Wise Comparison of AWS and GCP Solutions on ICDS$_R$ test set

such as 'pm,' 'days,' and 'mgdl,' and bigrams and trigrams like '10 days,' 'daily 10,' and 'cr x ray,' suggesting an orientation towards experimental or lab-result oriented narratives, with a particular emphasis on procedural timelines and diagnostic procedures. Hence, ICDS$_R$ focuses on a broader scope involving diagnostics and treatment monitoring.

Real versus Generated Datasets

ICDS^g_G vs n2c2-2006: We analyzed the n2c2-2006 and the synthetic ICDS^g_G discharge summaries in terms of summary statistics, Jaccard distance, and BERTScore (using the "dmis-lab/biobert-v1.1" model) as shown in Table 4 (Lee et al., 2020; Zhang et al., 2020). The Jaccard distance suggests a high level of lexical dissimilarity between the datasets, indicating that the synthetic dataset introduces a significant degree of variation compared to the real dataset. While indicating some differences, an F1 score of 0.6362 indicates the real and synthetic datasets have semantic overlap. An n-gram analysis of the top 10 unigrams, bigrams, and trigrams unveils the differences between the two datasets, yet also underscores their relevance to the task at hand as shown in App. Fig.14, Fig.15, Fig.16, and Fig.17. These metrics suggest that while the synthetic dataset is designed to be distinct enough to introduce useful variability, it retains a substantive semantic similarity to the real dataset. This balance is crucial when synthetic data is used for tasks such as model training, where the goal is to ensure that the model is not only trained on a diverse set of data but also remains relevant and effective when applied to real-world data. The high Jaccard distance combined with the moderate BERTScore indicates that the synthetic dataset achieves this objective by being similar enough to the real dataset to be useful, yet different enough to enhance the dataset's diversity and robustness.

ICDS^l_G vs ICDS$_R$: Similar to the n2c2-2006 and ICDS^g_G datasets, we analyzed the ICDS$_R$ and ICDS^l_G datasets with summary statistics, Jaccard distance, and BERTScore, as shown in Table 5. The Jaccard distance suggests lexical dissimilarity implying injection of new vocabulary in the generated discharge summaries. The n-gram analysis of the top 10 unigrams, bigrams, and trigrams shows these differences (App. Fig.18, Fig.19, Fig.20, and Fig.21). The BERTScore results indicate a moderate level of semantic similarity between the real and generated datasets. The metrics suggest that the generated dataset has greater lexical variety and incorporates some additional semantic constructs.

Evaluation of The Quality of Generated Summaries: The confusion matrix on convenience sample of 60 discharge summaries evaluated by physician1 and physician2 are shown in Fig. 2 and Fig. 3 respectively. There are 10 summaries that were originally synthetic but were labeled as real by physician 1, and 19 summaries that were originally synthetic but were labeled as real by physician 2. Physician 1 is able to label summaries with higher precision and recall, i.e., higher f1-score as com-

Physician	Precision	Recall	F1-score
Physician 1	0.714	0.833	0.769
Physician 2	0.537	0.733	0.620

Table 11: Evaluation metrics of 60 discharge summaries annotated by physician 1 and physician 2

pared to physician 2 (Table 11). The Cohen's kappa coefficient, the measure of inter-annotator agreement, is 0.290 showing a fair agreement between the labels assigned by the physicians. Additionally, physician 1 observed that many of the discharge summaries that he labeled synthetic appeared to have been translated from a non-English source. Physician 2 reported some diagnosis and formatting issues among the summaries he labeled as synthetic. Additionally, physician 2 reported some errors in diagnoses, medications, and lab results, but these were not limited to the summaries he labeled as synthetic.

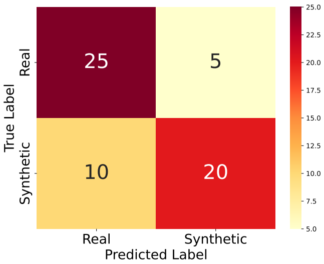


Figure 2: Confusion matrix on convenience sample (60 discharge summaries) evaluated by physician 1

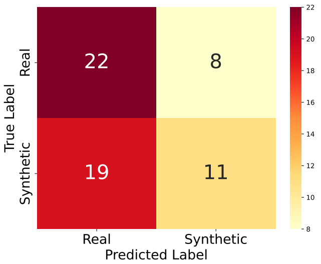


Figure 3: Confusion matrix on convenience sample (60 discharge summaries) evaluated by physician 2

Model Performance: Table 6 shows the results for intra- and inter-institutional performance. As can be observed, the inter-institutional performance of the model is very high (> 0.96 F1). However, the cross-institutional performance suffers significantly. Table 7 shows the results of training on generated datasets. The fine-tuned Model gives 68% F1 score on the ICDS$_R$ test set, 77% on the n2c2-2014 test set, and 85% on the n2c2-2006. Results on the ICDS$_R$ test set are not promising. This might have happened because ICDS$_G^g$ was generated using n2c2-2006. Fine-tuning on ICDS$_G^l$ dataset results in 98% F1 score on the ICDS$_R$ test set, 67% on n2c2-2014 test set, and 55% on the n2c2-2006 test set. To further improve model generalization, we experimented with combinations of datasets. Table 8 shows the training results on a combination of real and synthetic datasets. We get micro-F1 of 97% on n2c2-2014 and ICDS$_R$ test set given that we have included n2c2-2014 and ICDS$_G^l$ datasets in training, but the performance of the model (88%) is also notable on n2c2-2006 dataset. These results indicate that fine-tuning on the combination improves cross-institutional performance.

Analysis: Our experiments indicate models have poor cross-institutional generalization. We performed several experiments with n2c2-2006, n2c2-2014, ICDS$_G^g$, and ICDS$_G^l$ datasets, and their combinations. The general trend is that fine-tuned model performance degrades heavily in cross-dataset settings. At the individual entity level, the F1 score for the PATIENT entity is consistent for all the fine-tuned models. For the DOCTOR and DATE entities, the F1 scores of all the fine-tuned models are also consistent, except for when the model is trained on the n2c2-2006 dataset and tested on the n2c2-2014 dataset and ICDS$_R$ test sets. For the ID entity, all the fine-tuned models have consistent F1 scores, except for when the model is trained on ICDS$_G^g$, and tested on n2c2-2014 and ICDS$_R$ datasets. We noticed performance variance in the LOCATION, AGE, and CONTACT entities. This could be because the LOCATION can be any local address without a specific format. AGE is either a number like '78 Y' or a word representation of that number like 'Seventy-Eight year old'. In most cases in the datasets, these types of words or tokens are tagged as OTHERS, and they are highly prevalent. This could be why the AGE tag was incorrectly predicted as OTHERS in cross-dataset settings. The entity CONTACT includes email, IP address, phone number, landline number, etc. However, their distribution is not uniform.

Our main aim was to develop a robust model that could de-identify medical text from Indian Healthcare Institutes. This was done by fine-tuning PI-RoBERTa on ICDS$_G^l$ where we are getting state-of-the-art performance on ICDS$_R$. Almost all the entities were correctly identified, with a few exceptions. A few PHI entities were misidentified with non-PHI entities (i.e., OTHERS) and vice versa, as can be seen in App. Fig. 26. However, the percentage of incorrect prediction is significantly less when considering the total support set of ICDS$_R$ test set. However, this fine-tuned model was not generalizable when we tested it on the n2c2-2006 and n2c2-2014 test sets, as seen in the Table 7. For model generalizability, we fine-tuned PI-RoBERTa on n2c2-2014 , ICDS^g_G and ICDS^l_G, tested on the n2c2-2006 test set. The results shown in Table 8 indicate that models are generalizing when we fine-tuned them on different combinations of datasets, although the F1 score for all entities is not consistent, as can be seen in App. Fig. 28a. Confusion matrix for all the experiments are shown in App. Fig. 22 Fig. 23 , Fig. 24 , Fig. 25 , Fig. 26 Fig. 27 , Fig. 28b.

Comparison with Commercial De-Identification Systems: The results obtained using AWS and GCP solutions are summarized in Table 9 and Table 10. The results clearly indicate that AWS and GCP do not perform well on ICDS$_R$ test set. This could be because systems have been trained on non-Indian specific clinical data. This underscores the importance of ensuring that de-identification caters to diverse demographics, which is essential for ensuring the efficacy and ethical deployment of these solutions.

The underperformance of commercial solutions in classifying PHI in ICDS$_R$ can be attributed to misidentification. Medical entities are mistaken as NAME/ LOCATION, while Pin-codes as ID. Names like 'Alia' and 'Adah' are not being consistently recognized as NAME by AWS and GCP. Patient IDs that start with CRNO: ########### or ADM-########## are not identified as PHI; these solutions probably aren't sure what CRNO, ADM stand for. 'B/O Kanav Viswanathan' is misidentified, where 'Kanav Viswanathan' is a name and B/O stands for Baby of but gets labeled as a LOCATION. 'Urvi Bhamini Faiyaz Kakar' is identified as Name by GCP but not by AWS. 'Wockhardt Hospitals,' hospital name was not identified as PHI. Medical terms like 'BILIRUBIN,' 'MALLOY EVELYN,' 'CR X Ray' and 'SERUM LIPASE' are misidentified as NAME when they describe medical tests. Similarly, 'CREATININE (M - JAFFE COMPENSATED)' is a medical test and 'JAFFE' is misidentified as NAME. 'Meropenem,' an antibiotic, is misidentified as NAME. Even terms like 'Ward' from room names such as 'Ward-B' occasionally get misidentified as NAME. Test results like '136/94mmHg' or 'TSH - 5.45' are misidentified as ID. Locations like 'Subramaniam Chowk' and 'Yohannan Nagar,' are also misidentified as NAME. Additionally, using GCP or AWS for PHI detection introduces variability, causing results to vary with each execution. These factors underscore the need for precision and consistency in data handling to mitigate performance issues in medical contexts.

De-identification using LLMs: We also conducted experiments of de-identifying clinical summaries using LLMs directly. A precision score of 0.55 was obtained. However, the model faced challenges in terms of recall. The recall scores were merely 0.11. We also evaluated the performance of Mistral-7B-Instruct-346v0.1 and Gemma. Surprisingly, the results obtained from these models were far inferior to those of Meta-Llama-3-8B-Instruct. Results suggest that the LLMs struggle to detect PHI in Indian medical discharge summaries.

8 Conclusion and Future Directions

In this paper, we explored the task of de-identification on Indian clinical discharge summaries. Experiments indicate a poor generalization of fine-tuned (on public datasets) models and poor performance of the off-shelf commercial systems. Experiments with LLM generated summaries look promising; the model fine-tuned on generated summaries and public datasets shows good generalization performance. Our results are based on a small test set. Using the insights from our work, we aim to set-up an active learning workflow that combines our fine-tuned model and human annotators to produce a larger test dataset on which we may evaluate overall model performance as well as by conditioning on a medical specialty. The augmented (generated summaries with original data) institution-specific dataset can be used to fine-tune NER models that have been pre-trained on PHI data cost-effectively. Achieving cross-institution portability remains a topic of active research. However, many open-source large language models can be deployed on-premise and, as described above, fine-tuned to provide an immediate and effective solution to personal data protection in Indian healthcare institutions.

9 Acknowledgements

We would like to thank Dr. Uttam Singh, Dr Prabhakar Mishra, and Dr Amit Goel for their support for this work.

References

Emily Alsentzer, John R Murphy, Willie Boag, Wei-Hung Weng, David Jindi, Tristan Naumann, and Matthew B McDermott. 2019. Publicly available clinical bert embeddings. *arXiv preprint arXiv:1904.03323*.

Amazon Web Services. Amazon comprehend medical documentation: Analyzing protected health information (phi) [online].

BCHEM4 Modified BioBERT. ghadeermobasher/BCHEM4-Modified-BioBERT-v1 · Hugging Face — huggingface.co [online].

Mukund Chaudhry Chaudhry, Arman Kazmi, Shashank Jatav, Akhilesh Verma, Vishal Samal, Kristopher Paul, and Ashutosh Modi. 2022. Reducing inference time of biomedical ner tasks using multi-task learning. In *Proceedings of the 19th International Conference on Natural Language Processing (ICON)*, pages 116–122.

Richard J Chen, Ming Y Lu, Tiffany Y Chen, Drew FK Williamson, and Faisal Mahmood. 2021. Synthetic data in machine learning for medicine and healthcare. *Nature Biomedical Engineering*, 5(6):493–497.

Edward Choi, Sushant Biswal, Bradley Malin, Jon Duke, Walter F Stewart, and Jimeng Sun. 2017. Generating multi-label discrete patient records using generative adversarial networks. *arXiv preprint arXiv:1703.06490*.

Clinical AI Apollo. Clinical-AI-Apollo/Medical-NER · Hugging Face — huggingface.co [online].

Franck Dernoncourt, Ji Young Lee, Özlem Uzuner, and Peter Szolovits. 2017. De-identification of patient notes with recurrent neural networks. *Journal of the American Medical Informatics Association*, 24(3):596–606.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Margaret Douglass, Gari D Clifford, Andrew Reisner, George B Moody, and Roger G Mark. 2004. Computer-assisted de-identification of free text in the mimic ii database. In *Computers in Cardiology, 2004*, pages 341–344. IEEE.

Dslim bert base NER. dslim/bert-base-NER · Hugging Face — huggingface.co [online].

Mohamed El Azzouzi, Gouenou Coatrieux, Reda Bellafqira, Denis Delamarre, Christine Riou, Naima Oubenali, Sandie Cabon, Marc Cuggia, and Guillaume Bouzillé. 2023. Automatic de-identification of french electronic health records: a cost-effective approach exploiting distant supervision and deep learning models. *BMC Medical Informatics and Decision Making*, 23(1):22.

European Parliament and Council of the European Union. General data protection regulation [online].

Daniele Faraglia and Other Contributors. 2010. Faker.

Andre Goncalves, Paroma Ray, Brendon Soper, et al. 2020. Generation and evaluation of synthetic patient data. *BMC Medical Research Methodology*, 20(1):108.

Google Cloud. Deidentify sensitive data [online].

Imanol Isasa, Mikel Hernandez, Gorka Epelde, et al. 2024. Comparative assessment of synthetic time series generation approaches in healthcare: leveraging patient metadata for accurate data synthesis. *BMC Medical Informatics and Decision Making*, 24(1):27.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Shreyasi Mandal and Ashutosh Modi. 2024. Iitk at semeval-2024 task 2: Exploring the capabilities of llms for safe biomedical natural language inference for clinical trials. *arXiv preprint arXiv:2404.04510*.

Stephanie M Meystre, Olivia Ferrandez, Jeff J Friedlin, Brett R South, Shuying Shen, and Matthew H Samore. 2010. Text de-identification for privacy protection: a study of its impact on clinical text information content. *Journal of the American Medical Informatics Association*, 17(1):19–24.

Ministry of Electronics and Information Technology (MeitY), India. Indian digital personal data protection act 2023 [online].

Ministry of Health and Family Welfare (MoHFW), India. National resource centre for ehr standards (nrces) [online].

Hiroki Nakayama, Takahiro Kubo, Junya Kamura, Yasufumi Taniguchi, and Xu Liang. 2018. doccano: Text annotation tool for human. Software available from https://github.com/doccano/doccano.

Ishna Neamatullah, Margaret M Douglass, Li-wei H Lehman, Andrew T Reisner, Mauricio Villarroel, William J Long, Peter Szolovits, George Moody, Roger G Mark, and Gari D Clifford. 2008. Automated de-identification of free-text medical records. *BMC medical informatics and decision making*, 8(1):32.

PI-RoBERTa. Roberta-ner-personal-info [online].

PIB Press Release. Pib press release [online].

Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.

Sunil Kumar Srivastava. 2016. Adoption of electronic health records: a roadmap for india. *Healthcare informatics research*, 22(4):261.

Amber Stubbs, Christopher Kotfila, and Özlem Uzuner. 2015. Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/uthealth shared task track 1. *Journal of Biomedical Informatics*, 58:S11–S19.

Latanya Sweeney. 2013. Matching known patients to health records in washington state data. *arXiv preprint arXiv:1307.1370*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models (2023). *arXiv preprint arXiv:2302.13971*.

U.S. Department of Health and Human Services (HHS). Health insurance portability and accountability act [online].

Matthijs J Warrens. 2015. Five ways to look at cohen's kappa. *Journal of Psychology and Psychotherapy*, 5.

Xi Yang, Tianchen Lyu, Qian Li, Chih-Yin Lee, Jiang Bian, William R Hogan, and Yonghui Wu. 2019. A study of deep learning methods for de-identification of clinical notes in cross-institute settings. *BMC Medical Informatics and Decision Making*, 19(1):60.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Özlem Uzuner, Isaac Goldstein, Yuan Luo, and Isaac S. Kohane. 2008. Identifying patient smoking status from medical discharge records. *Journal of the American Medical Informatics Association*, 15(1):14–24.

Appendix

Table of Contents

List of Tables

List of Figures

A Prompts and Synthetic Discharge Summaries

In Table 12, we showcase the prompts which we used to generate the ICDS_G^g and ICDS_G^l datasets. We used Prompt A in Table 12 for generating ICDS_G^g from Gemini-pro-1.0. Table 13 gives a sample discharge summary. Using prompt B in Table 12, we generated ICDS_G^l dataset using Llama-3-8B-Instruct. Table 14 gives a sample discharge summary.

B Tag Mapping across all the dataset and tag Distribution after Mapping the Tags

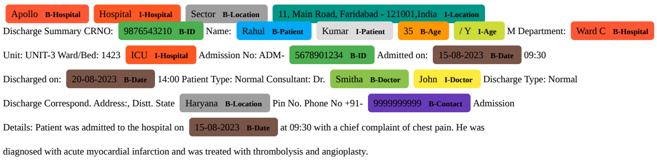
We have five datasets: n2c2-2006, n2c2-2014, ICDS_R, ICDS_G^g, and ICDS_G^l. Each dataset has its own tag set. n2c2-2006 contains 9 tags, n2c2-2014 contains 24, ICDS_R contains 26, ICDS_G^g contains 34, and ICDS_G^l contains 106 unique tags, including the OTHERS tag. In the datasets n2c2-2006, n2c2-2014, and ICDS_R, all the tags are related to PHI entities. However, in the ICDS_G^l and ICDS_G^g datasets, a few annotated tags are not related to the PHI entities due to LLM hallucinations. To train models for a fair comparison, we need a uniform tag set across all datasets.

Hence, we mapped the tag set of all the datasets to the n2c2-2006 tag set. In all the datasets, we mapped entities like street, city, country, zip, etc to LOCATION. Similarly, we mapped phone number, mobile number, email, landline, etc, to CONTACT. Additionally, we mapped all the PHI-related entities to their super-set using mapping shown in Table 15. In the ICDS_G^l and ICDS_G^g datasets, we have several tags unrelated to the PHI entities. Hence, we mapped all non-PHI entities to the OTHERS tag. After mapping the tag set of all the datasets to n2c2-2006 tag set, we calculated the tag distribution of all PHI entities across all datasets. The distribution of tag sets of all the dataset when mapped with n2c2-2006 dataset are shown in Fig. 5, Fig. 6, Fig. 7, Fig. 8, Fig. 9, Fig. 10, Fig. 11, and Fig. 12.

C Corpus Statistics

The n-gram frequencies from the n2c2-2006 dataset show a strong emphasis on clinical and procedural language, including terms like ‘mg,’ ‘po,’ and ‘hospital,’ as shown in Fig. 14. Notably, phrases such as ‘discharge summary’ and ‘physical examination’ dominate, highlighting standard documentation practices. Trigrams such as ‘dis report status’ and ‘report status unsigned’ indicate typical phrasing in medical reports. This is in contrast with the ICDS_R dataset in Fig. 18, where there is a predominance of time-related unigrams (‘pm,’ ‘days’) and clinical terms (‘mgdl,’ ‘method’). The frequent bigrams and trigrams revolve around treatment and diagnosis descriptors, such as ‘daily 10 days’ and ‘x ray chest,’ illustrating the detailed recording of patient care routines and diagnostic procedures commonly found in medical records. In the n2c2-2006 dataset, bigrams like ‘mg po’ and ‘discharge date,’ and trigrams like ‘mg po bid’ and ‘history present illness,’ which reveal specific medication dosages and detailed descriptions of patient conditions, are found next to PHI elements, as shown in Fig. 16. In the ICDS_R dataset, specific trigrams like ‘discharge summary crno’ and ‘normal discharge correspond’ are located near PHI elements (Fig. 20). The differences between the n2c2 2006 dataset and ICDS_R highlight how clinical documentation practices and language differ between the US and India.

In the synthetic ICDS_G^g dataset, the frequent occurrence of ‘phi’ in various n-grams highlights (in Fig. 15) the inclusion of potentially identifiable information. Trigrams such as ‘phi typehospitalfihphi’ and ‘phi typeid7673299w3phi’ illustrate the use of placeholders for personal identifiers, indicative of the synthetic nature of the dataset and its focus on mimicking real-world PHI data while maintaining privacy. In the ICDS_G^l dataset, the frequent mention of basic terms like ‘patient,’ ‘discharge,’ and ‘history’ reflects their regular usage in clinical documents, as seen in Fig. 19. Phrases such as ‘discharge summary’ and ‘medical history’ indicate standardized document formats. For n-grams next to PHI elements in the synthetic ICDS_G^g dataset as seen in Fig. 17, we observe a mix of clinical terminology (‘discharge,’ ‘patient,’ ‘history’) and documentation descriptors (‘text record,’ ‘reportend text’). Bigrams and trigrams like ‘discharge summary patient’ and ‘text record record’ suggest a replication of typical medical documentation formats. Terms like ‘primary care physician’ and ‘history present illness’ reflect the comprehensive nature of clinical narratives. In contrast, the n-grams next to PHI elements in the ICDS_G^l dataset, as shown in Fig. 21, highlight the frequent use of both temporal (‘pm’, ‘days’) and medical (‘mgdl,’ ‘discharge’) terms. Common bigrams and trigrams such as ‘discharge summary,’ ‘cr x ray,’ and ‘x ray chest’ underscore the clinical focus on diagnostic imaging and summary documentation. The



Apollo B-Hospital Hospital I-Hospital Sector B-Location 11, Main Road, Faridabad - 121001,India I-Location
Discharge Summary CRNO: 9876543210 B-ID Name: Rahul B-Patient Kumar I-Patient 35 B-Age / Y I-Age M Department: Ward C B-Hospital
Unit: UNIT-3 Ward/Bed: 1423 ICU I-Hospital Admission No: ADM- 5678901234 B-ID Admitted on: 15-08-2023 B-Date 09:30
Discharged on: 20-08-2023 B-Date 14:00 Patient Type: Normal Consultant: Dr. Smitha B-Doctor John I-Doctor Discharge Type: Normal
Discharge Correspond. Address:, Distt. State Haryana B-Location Pin No. Phone No +91- 9999999999 B-Contact Admission
Details: Patient was admitted to the hospital on 15-08-2023 B-Date at 09:30 with a chief complaint of chest pain. He was
diagnosed with acute myocardial infarction and was treated with thrombolysis and angioplasty.

Figure 4: Pre-processed Discharge Summary after adding B and I tags

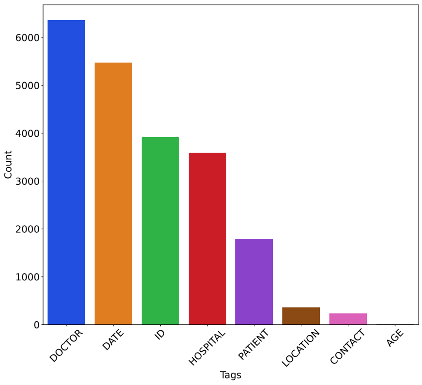


Figure 5: Tag Distribution in n2c2-2006 train dataset

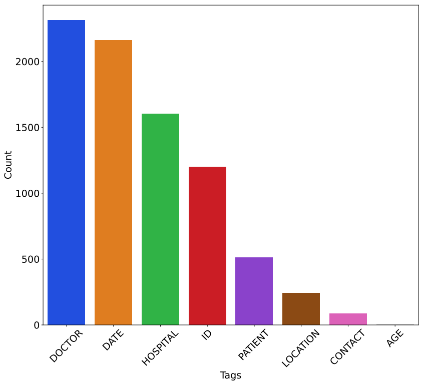


Figure 6: Tag Distribution in n2c2-2006 test dataset

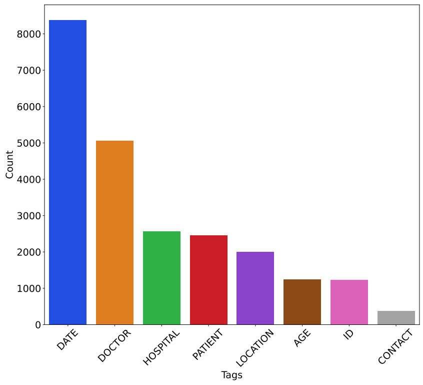


Figure 7: Tag Distribution in n2c2-2014 train dataset

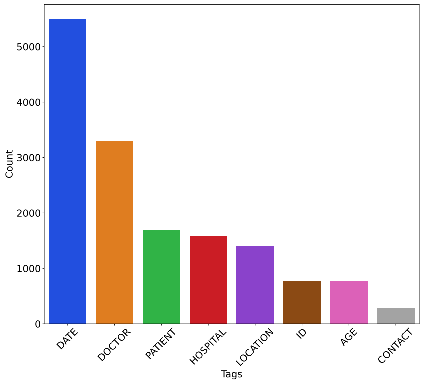


Figure 8: Tag Distribution in n2c2-2014 test dataset

trigrams involving ‘daily 10 days’ and ‘x ray chest bed’ reflect specific medical interventions and patient care protocols typically documented in patient records.

D Model Training Details

We fine-tuned dslim/bert-base-NER (Dslim bert base NER), ghadeermobasher/BCHEM4-Modified-BioBERT-v1 (BioBERT), and Clinical-AI-Apollo/Medical-NER (Clinical AI Apollo). We obtained a consistent train-set F1 Score for PHI entities from these models after fine-tuning, but the performance of these models decreased significantly when we tested them on cross-dataset settings. However, after fine-tuning, PI-RoBERTa outperformed these models in the same and cross-dataset settings, so we chose PI-RoBERTa for further experiments. Fig. 13 shows the model architecture.

PI-RoBERTa was fine-tuned on each training set as given in Table 3 and tested on each corresponding test set. We fixed the hyperparameters for all the experiments. The model was fine-tuned at four epochs in all the experiments with a batch size of 8; the learning rate was 5e-5. We used Weighted Cross entropy loss to handle the data imbalance problem because around 90 percent of the tokens correspond to non-PHI entities in all datasets. After several experiments, we devised a formula to assign weights to different Entities. $w_t = \log\left(4 \times \frac{n}{n_t}\right)$, where w_t is the weight assigned to the t^{th}entity; n_t is the number of tokens in the t^{th} entity; n is the total number of tokens in the dataset

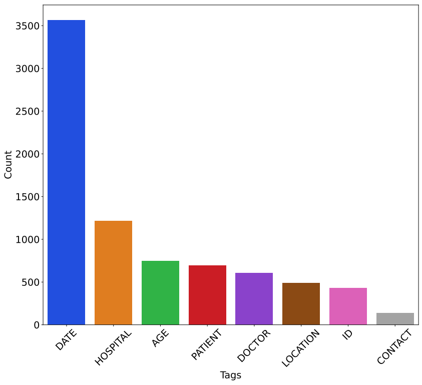


Figure 9: Tag distribution in ICDS$_R$ train dataset

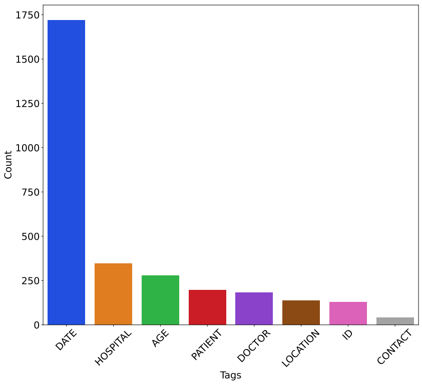


Figure 10: Tag distribution in ICDS$_R$ test dataset

E Evaluation Metrics

Model was evaluated using various performance metrics as described below.

- Macro Precision:

$$\text{Precision}_{\text{macro}} = \frac{1}{n}\sum_{i=1}^{n}\frac{TP_i}{TP_i + FP_i}$$

- Macro Recall

$$\text{Recall}_{\text{macro}} = \frac{1}{n}\sum_{i=1}^{n}\frac{TP_i}{TP_i + FN_i}$$

- Macro F1-score

$$\text{F1-score}_{\text{macro}} = \frac{2 \times \text{Precision}_{\text{macro}} \times \text{Recall}_{\text{macro}}}{\text{Precision}_{\text{macro}} + \text{Recall}_{\text{macro}}}$$

- Micro Precision:

$$\text{Precision}_{\text{micro}} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n}(TP_i + FP_i)}$$

- Micro Recall

$$\text{Recall}_{\text{micro}} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n}(TP_i + FN_i)}$$

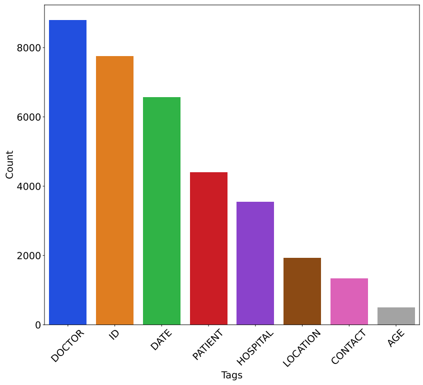


Figure 11: Tag distribution in ICDS^g_G dataset

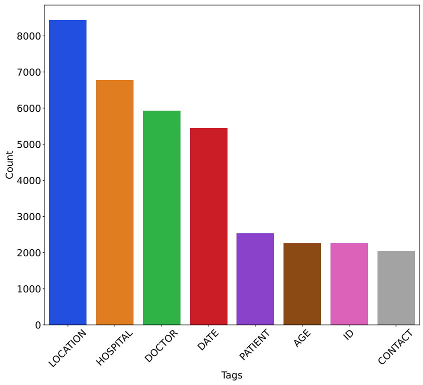


Figure 12: Tag distribution in ICDS^l_G dataset

- Micro F1-score

$$\text{F1-score}_{\text{micro}} = \frac{2 \times \text{Precision}_{\text{micro}} \times \text{Recall}_{\text{micro}}}{\text{Precision}_{\text{micro}} + \text{Recall}_{\text{micro}}}$$

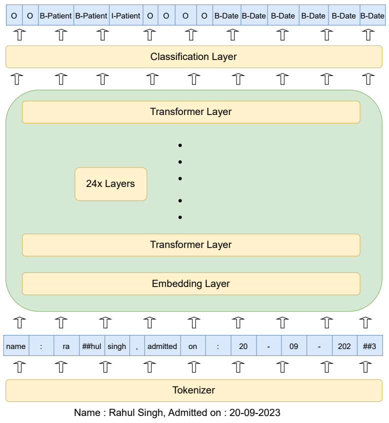


Figure 13: Architecture of PI-RoBERTa

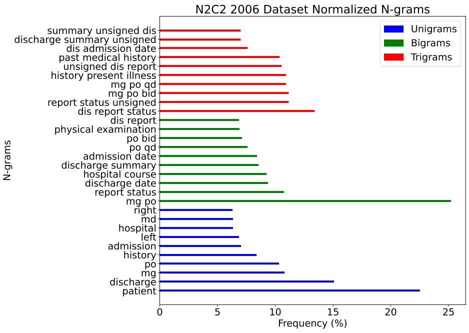


Figure 14: n2c2-2006 Top 10 N-grams

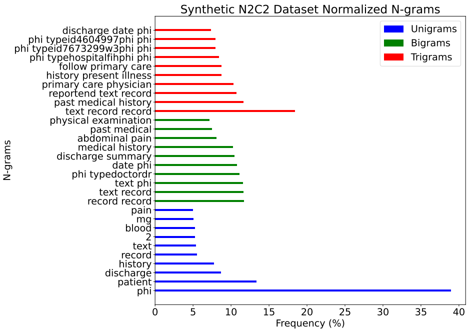


Figure 15: ICDS$^{g}_{G}$ Top 10 N-grams

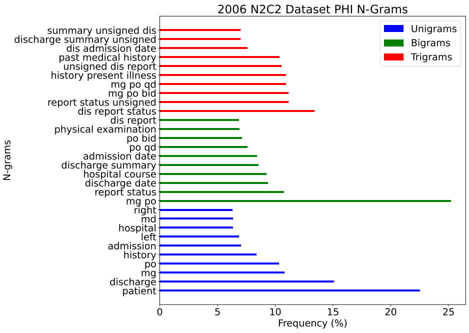


Figure 16: n2c2-2006 Top 10 N-grams Spanning PHI Elements

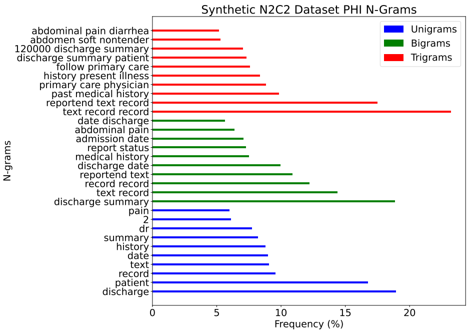


Figure 17: ICDS$^{g}_{G}$ Top 10 N-grams Spanning PHI Elements

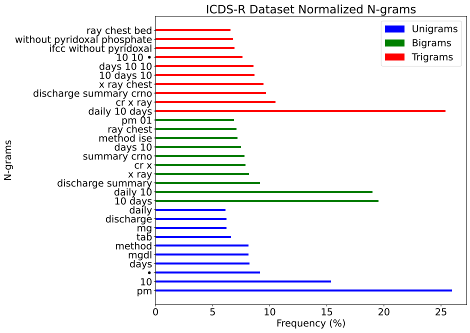


Figure 18: ICDS$_{R}$ Top 10 N-grams

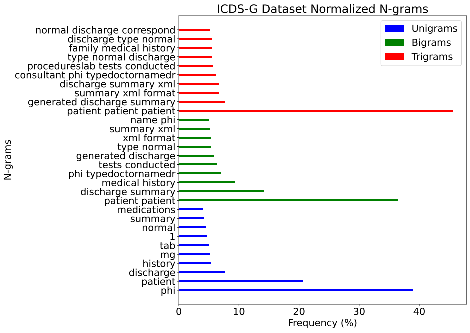


Figure 19: ICDS$^{l}_{G}$ Top 10 N-grams

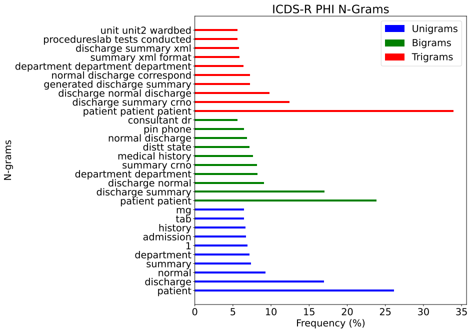


Figure 20: ICDS$_{R}$ Top 10 N-grams Spanning PHI Elements

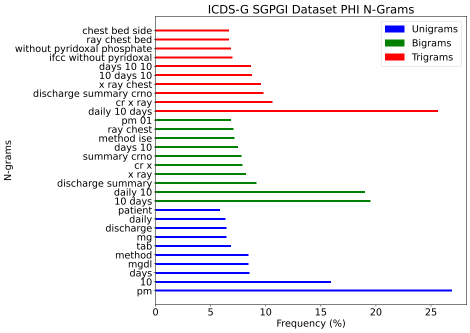


Figure 21: ICDS$^{l}_{G}$ Top 10 N-grams Spanning PHI Elements

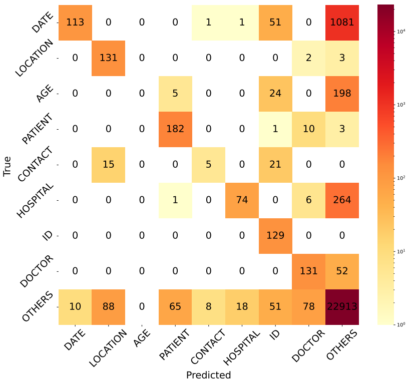


Figure 22: Confusion matrix on ICDS$_R$ test set when PI-RoBERTa finetuned on n2c2-2006

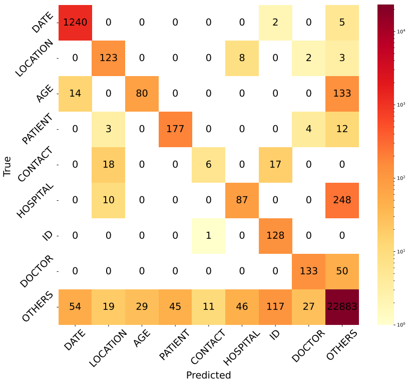


Figure 23: Confusion matrix on ICDS$_R$ test set when PI-RoBERTa finetuned on n2c2-2014

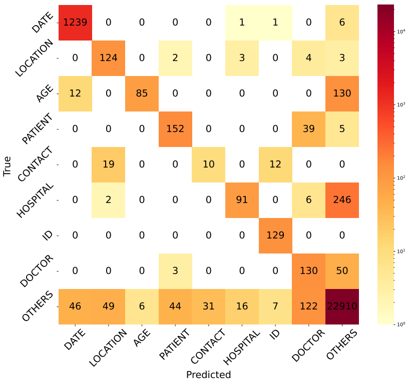


Figure 24: Confusion matrix on ICDS$_R$ test set when PI-RoBERTa finetuned on Combining n2c2-2006 and n2c2-2014

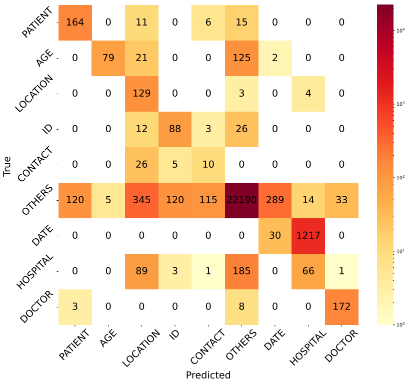


Figure 25: Confusion matrix on ICDS$_R$ test set when PI-RoBERTa finetuned on ICDS^g_G

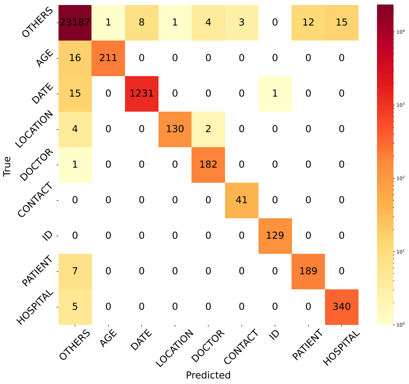


Figure 26: Confusion matrix on ICDS$_R$ test set when PI-RoBERTa finetuned on ICDS^l_G

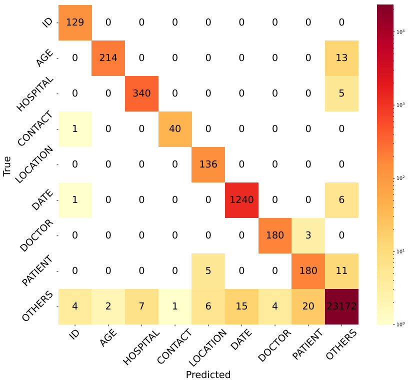
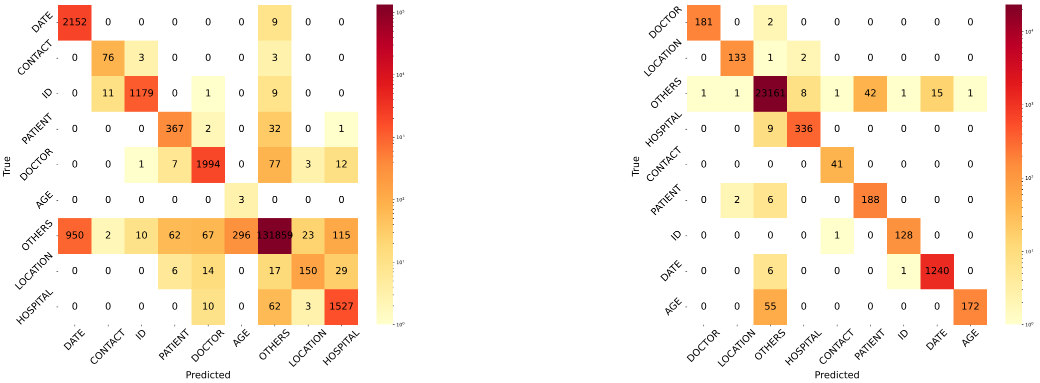


Figure 27: Confusion matrix on ICDS$_R$ test set when PI-RoBERTa finetuned on Combining ICDS^l_G and ICDS^g_G dataset



	DATE	CONTACT	ID	PATIENT	DOCTOR	AGE	OTHERS	LOCATION	HOSPITAL
DATE	2152	0	0	0	0	0	9	0	0
CONTACT	0	76	3	0	0	0	3	0	0
ID	0	11	1179	0	1	0	9	0	0
PATIENT	0	0	0	367	2	0	32	0	1
DOCTOR	0	0	1	7	1994	0	77	3	12
AGE	0	0	0	0	0	3	0	0	0
OTHERS	950	2	10	62	67	296	131859	23	115
LOCATION	0	0	0	6	14	0	17	150	29
HOSPITAL	0	0	0	0	10	0	62	3	1527

(a) Confusion matrix on n2c2-2006 test set when PI-RoBERTa finetuned on combining n2c2-2014, ICDS^g_G, and ICDS^l_G dataset.

	DOCTOR	LOCATION	OTHERS	HOSPITAL	CONTACT	PATIENT	ID	DATE	AGE
DOCTOR	181	0	2	0	0	0	0	0	0
LOCATION	0	133	1	2	0	0	0	0	0
OTHERS	1	1	23161	8	1	42	1	15	1
HOSPITAL	0	0	9	336	0	0	0	0	0
CONTACT	0	0	0	0	41	0	0	0	0
PATIENT	0	2	6	0	0	188	0	0	0
ID	0	0	0	0	1	0	128	0	0
DATE	0	0	6	0	0	0	1	1240	0
AGE	0	0	55	0	0	0	0	0	172

(b) Confusion matrix on ICDS$_R$ test set when PI-RoBERTa finetuned on combining ICDS^l_G, ICDS^g_G, n2c2-2006, and n2c2-2014 dataset.

Figure 28: Confusion matrix on n2c2-2006 and ICDS$_R$ testset when PI-RoBERTa fine-tuned on combination of generated and real data.

Prompt Id	Prompt
A	Generate discharge summaries for Indian patients, capturing the essence of healthcare in India. The summaries should integrate conventional medical treatments with traditional remedies, reflecting the holistic approach embraced by Indian healthcare systems. Incorporate prevalent Indian health conditions, treatments, and culturally relevant follow-up care instructions. To ensure authenticity, each summary should include distinct patient details like name, age, address, contact information, hospital, doctor, and ID. Include prevalent diseases in India such as Tuberculosis (TB), Diabetes, Cardiovascular Diseases, Respiratory Infections, Hypertension, Dengue Fever, Malaria, Hepatitis, Chronic Kidney Disease (CKD), Cancer, Typhoid Fever, Cholera, HIV/AIDS, Japanese Encephalitis, Leptospirosis, Rabies, Tuberculosis of the Central Nervous System (CNS TB), Rheumatic Heart Disease, Iron-Deficiency Anemia, and Chikungunya. Also, laboratory test reports of the chosen disease should be included. Ensure the format of generated discharge summaries is similar to the summary given in the prompt, i.e., in XML format. Example: Patient Summary: <discharge summary> Generate the summaries that have a minimum of 2048 words. Ensure there is consistent consistency between the doctor's name, patient name, drug-disease, etc.
B	Generate an extensive discharge summary of at least 2048 words tailored for Indian patients. To ensure authenticity, the generated summary must include distinct patient-specific details like name, age, address, contact information, hospital name, doctor name, and unique ID. Maintain coherence across all the elements, doctor's name, patient's identity, medications, diseases, etc. Ensure all the PHI (personal health information) elements are properly annotated to maintain privacy and authenticity. The generated discharge summary should be XML-formatted with PHI annotations. The generated summaries should include following sections: Admission Details, Diagnosis / Chief Complaints, Allergies, Physical Examination, Medical History, Family Medical history, Treatment Plan, Investigations, Medications (List of medications prescribed at discharge), Follow-up Instructions, Procedures/Lab Tests Conducted (List of procedures or tests conducted during hospital stay, along with results if available), and Special Instructions. Please ensure that these sections are incorporated into the generated summaries, but refrain from including them as tags in the output. The generated summary should be properly enclosed within the <RECORD> and </RECORD> tags to ensure it's within the XML format. Here's an example patient summary: Patient Summary: <discharge summary>
C	Generate an extensive synthetic discharge summary of at least 2048 words tailored for Indian patients. Generated summary must include distinct entities like name, age, address, contact information, hospital name, doctor name, and unique ID. Maintain coherence across all the elements, doctor's name, patient's identity, medications, diseases, etc. Identify all entities in the generated text and mark these with XML tags as in the following example:<TYPE='Insurance Number'>AB123456C</TYPE> entities= ['Patient Name','Hospital_Name','Staff_Name','Doctor_Name','Age','Gaurdian_Name','Gender', 'Patient_ID','Misc_Medical_ID','Aadhar','Driver_License','Voter_ID','PAN_Card','Patient_DOB', 'Treatment_Date','Treatment_Time','Phone_No','Landline','Email','IP_Address','Fax','Doctor_Specialisation', 'Patient_Profession','City','Ward_Location','Device_Number','Other_Info','State','Street','Zip','Country', 'Other_Location','Other_Govt_ID','Insurance_Number','Web_url'] The generated summaries should include following sections: Admission Details, Diagnosis / Chief Complaints, Allergies, Physical Examination, Medical History, Family Medical history, Treatment Plan, Investigations, Medications (List of medications prescribed at discharge), Follow-up Instructions, Procedures/Lab Tests Conducted (List of procedures or tests conducted during hospital stay, along with results if available), and Special Instructions. Please ensure that these sections are incorporated into the generated summaries, but refrain from including them as tags in the output. The generated summary should be properly enclosed within the <RECORD> and </RECORD> tags to ensure it's within XML format. Here's an example synthetic patient summary: <discharge summary>.

Table 12: Prompts used for synthetic discharge summary generation

Model Name	Generated Discharge Summary
Gemini-pro-1.0	829369 HODYEGE DISTRICT HOSPITAL 9151843 830690 408542 12/12/10/2023 23:59:00 PM Discharge Summary Discharge Report Status : Signed DISCHARGE SUMMARY NAME : NIVEDITHA CHETTY UNIT NUMBER : 393-29-55 ADMISSION DATE : 12/10/10/2023 DISCHARGE DATE : 12/12/10/2023 PRINCIPAL DIAGNOSIS : Prolapsed intervertebral disc with radiculopathy of L4-L5, L5-S1, L2-L3, with weakness of right lower limb. ASSOCIATED DIAGNOSIS : CVA (LEFT SIDED HEMIPARESIS) ADVERSE DRUG REACTIONS : None . CLINICAL INFECTIONS : None . PRINCIPAL PROCEDURE : Spinal surgery ASSOCIATED PROCEDURES : Physiotherapy . HISTORY OF PRESENT ILLNESS : Mrs. NIVEDITHA CHETTY is a 58 years old female who presented to the emergency department with complaints of low back pain for 1 year and weakness of right lower limb for 1 month . Weakness was insidious in onset and gradually increased to the present level . There was no numbness of lower limbs . No urinary or bowel involvement . No fever , cough , gastrointestinal symptoms or cardio-respiratory symptoms . PAST MEDICAL HISTORY : CVA (LEFT SIDED HEMIPARESIS) . SOCIAL HISTORY : Non smoker and non alcoholic . PHYSICAL EXAMINATION : VITALS : Temperature : 98.6\302\260 F, BP : 120/80 mmHg , SPO2 : 95% , Respiratory rate : 20/min , Pulse rate : 80/min CNS : GCS 15/15 , Cranial nerves - Normal . Power - Right lower extremity grade - 4/5 . Sensation - Intact . Motor system examination - Tone - Increased on the right side . Deep tendon reflexes - Brisk on the right side . Plantar - Extensor on the right side . CVS : No pallor , icterus , clubbing , pedal edema , cyanosis , SPO2 - 99% . JVP not raised . Carotid pulsations are well felt and equal on both the sides . No bruits audible . Heart sounds - Regular . No murmurs , gallops , rubs or extra sounds . Respiratory system : Chest expansion - Symmetrical . Air entry reduced bilaterally but more on the right side . No added sounds . Breath sounds decreased in right infrascapular area . Abdomen : Soft , non-tender , no mass or organomegaly . LABORATORY DATA : Hb - 12.0 gm/dl , TLC - 8,000/cmm , Platelets - 2.5 lacs/cmm , ESR - 20 mm/hr . Blood sugar - 100 mg/dl , Urea - 30 mg/dl , Creatinine - 1.0 mg/dl , Uric acid - 5.0 mg/dl . Liver function tests - Normal . Lipid profile - Normal . Electrolytes - Normal . X-RAY : LUMBAR SPINE AP LATERAL VIEW : There is narrowing of the L4-L5, L5-S1, L2-L3 intervertebral spaces with sclerosis of the adjacent end plates . Loss of lumbar lordosis with mild dextroscoliosis . CT SCAN : LUMBAR SPINE : There is a prolapsed intervertebral disc at L4-L5, L5-S1, L2-L3 with foraminal stenosis on the right side at L4-L5 . Thecal sac is compressed at L4-L5, L5-S1 due to central disc prolapse . MRI SCAN : LUMBAR SPINE : There is a prolapsed intervertebral disc at L4-L5, L5-S1, L2-L3 with foraminal stenosis on the right side at L4-L5 . Thecal sac is compressed at L4-L5, L5-S1 due to central disc prolapse . There is evidence of denervation of the right sciatic nerve and cauda equina . HOSPITAL COURSE : The patient was admitted for surgical intervention . After proper pre-operative evaluation the patient was taken up for surgery . During surgery , a decompression of the thecal sac and foramina was done . Postoperatively , the patient was started on physiotherapy . The patient was able to walk with the support of walker . MEDICATIONS ON DISCHARGE : Tramadol - 50 mg - BD SOS - 7 days Aceclofenac - 100 mg - BD - 7 days Multivitamin and minerals - OD FOLLOW UP : The patient will be reviewed in the clinic in 2 weeks . DR.GYANESHWARI JANGID MBBS, MD, MS (Orthopedics) HODYEGE DISTRICT HOSPITAL Electronically Signed DR.GYANESHWARI JANGID [report_end]

Table 13: Example summary generated using gemini-pro-1.0

Model Name	Generated Discharge Summary
Meta-Llama-3-8B-Instruct	Max Super Specialty Hospital Sector 11, Dwarka, New Delhi - 110075,India Discharge Summary CRNO: 1234567890 Name: Rahul Kumar 35/Y/M Department: Ward E Unit: UNIT-3 Ward/Bed: 5432 ICU Admission No: ADM-9012345678 Admitted on: 25-08-2023 14:30 Discharged on: 28-08-2023 10:00 Patient Type: Normal Consultant: Dr. Rohan Sharma Discharge Type: Normal Discharge Correspond. Address:, Distt. State Uttar Pradesh Pin No. Phone No +91-9812345678 FOLLOW UP: Appointment on 15/10/2023 Morning Block-Upto 09:00 A.M.INVESTIGATION RESULTS: 25/08/2023 09:30 AM Blood Urea 40 mg/dL 25/08/2023 09:30 AM Serum Creatinine 1.2 mg/dL 25/08/2023 09:30 AM Hb 12.5 g/dL Admission Details: Patient was admitted to the hospital with a diagnosis of Acute Kidney Injury (AKI) due to dehydration. He was a known hypertensive and diabetic patient. He presented with symptoms of fatigue, weakness, and decreased urine output. Diagnosis/Chief Complaints: Primary Diagnosis: Acute Kidney Injury (AKI) Secondary Diagnosis: Hypertension, Diabetes Mellitus Allergies: None reported Physical Examination: The patient was conscious and oriented. He had a pulse rate of 100 beats per minute, blood pressure of 160/100 mmHg, and respiratory rate of 20 breaths per minute. He had bilateral pedal edema and decreased urine output. Medical History: The patient had a history of hypertension and diabetes mellitus. He was taking medications for these conditions, including metformin and lisinopril. Family Medical History: The patient's father had a history of hypertension and heart disease. Treatment Plan: The patient was treated with intravenous fluids and medications to manage his symptoms. He was also started on dialysis to manage his acute kidney injury. Investigations: The patient underwent the following investigations: Blood Urea 40 mg/dL Serum Creatinine 1.2 mg/dL Hb 12.5 g/dL Medications: The patient was prescribed the following medications at discharge: Metformin 500 mg twice daily Lisinopril 10 mg daily Furosemide 20 mg daily Hydrochlorothiazide 25 mg daily Aspirin 75 mg daily Follow-up Instructions: The patient was advised to follow up with his primary care physician in two weeks for a check-up and to monitor his blood sugar and blood pressure levels. Procedures/Lab Tests Conducted: The patient underwent the following procedures and lab tests during his hospital stay: Blood tests: Blood Urea 40 mg/dL, Serum Creatinine 1.2 mg/dL, Hb 12.5 g/dL Urine tests: Urine output decreased, proteinuria Dialysis: Three sessions of dialysis were performed Special Instructions: The patient was advised to follow a strict fluid intake and to monitor his urine output closely. He was also advised to take his medications as prescribed and to follow up with his primary care physician regularly. Prepared by (Dr. Rakesh Kumar) Signature of Consultant Printed on 29-08-2023 14:30:15 Temp Employee Male @ 192.168.1.100 Page 2 / 2

Table 14: Example summary generated using llama-3-8B-Instruct

Original tags	Mapped Tags
Treatment_Date, Patient_DOB, Investigation_Date, Admission Date, Procedure_Date, Date	DATE
Ward_Location, Hospital_Name, Department	HOSPITAL
Patient_ID, Misc_Medical_ID, Employee_ID, Admission Number	ID
Age	AGE
Doctor_Name, Staff_Name, Prepared by, Signature, Doctor_Signature, Signature of Consultant	DOCTOR
Patient_Name, Gaurdian_Name, Patient_Signature, Patient_Spouse, Family_Member_Name	PATIENT
Zip, Phone_No, Landline, IP_Address, Phone, Contact_Info, Contact_Number, Contact_No, Mobile, Phone Number, Patient_Phone, Email, Email_ID, Contact Information, Phone No	CONTACT
City, State, Country, Street, Other_Location, Correspondence_Address, Contact_Address, Contact Information, Pin, Pin Code, Pin_No, Postal_Code, Address, Contact_Address	LOCATION

Table 15: Tag mapping from PHI entities in the different datasets to the PHI entity set of n2c2-2006 dataset, and all other non-PHI entities are mapped with Others tag

Pre-training data selection for biomedical domain adaptation using journal impact metrics

Mathieu Laï-king and **Patrick Paroubek**
Université Paris-Saclay, CNRS,
Laboratoire Interdisciplinaire des Sciences du Numérique,
91400, Orsay, France
{mathieu.laiking,patrick.paroubek}@lisn.upsaclay.fr

Abstract

Domain adaptation is a widely used method in natural language processing (NLP) to improve the performance of a language model within a specific domain. This method is particularly common in the biomedical domain, which sees regular publication of numerous scientific articles. PubMed, a significant corpus of text, is frequently used in the biomedical domain. The primary objective of this study is to explore whether refining a pre-training dataset using specific quality metrics for scientific papers can enhance the performance of the resulting model. To accomplish this, we employ two straightforward journal impact metrics and conduct experiments by continually pre-training BERT on various subsets of the complete PubMed training set, we then evaluate the resulting models on biomedical language understanding tasks from the BLURB benchmark. Our results show that pruning using journal impact metrics is not efficient. But we also show that pre-training using fewer abstracts (but with the same number of training steps) does not necessarily decrease the resulting model's performance.

1 Introduction

Advances in deep learning for natural language processing (NLP) in recent years have enabled transfer learning to develop (Ruder et al., 2019), particularly since the creation of *Transformers* (Vaswani et al., 2017).

One type of transfer learning aims to start with a pre-training phase where the model learns the general language structure and then a second phase where the model can be fine-tuned for a specific task. In the context of deep learning for NLP, this method avoids re-training a model from scratch for each new task, starting with a model that already has general language knowledge. These pre-trained models generally use a large corpus of text.

A specialized domain, such as finance or the biomedical domain, may contain numerous tasks. In the case of language, a specialized domain has a specific vocabulary containing terms more rarely found in general texts. We can observe this phenomenon when looking at tokens produced by a biomedical tokenizer against a general tokenizer (Boukkouri et al., 2022). Moreover, tasks may require domain-specific knowledge not found in general sources. So, to improve the performance of a model previously trained on a general domain to a specific domain, it is interesting to use a corpus specific to the domain to which we wish to adapt our model.

Most of the data used for pre-training in the biomedical field are research articles and papers that can be either abstracts, full texts, or a combination of both. This data generally originates from large public databases such as PubMed or PubMedCentral (for full-text articles). However, to our knowledge, no study has examined the selecting subsets of these large databases for pre-training using metrics specific to scientific papers. That leads us to our research question: Can a language model be adapted to the biomedical domain by efficiently selecting scientific documents in the pre-training data while maintaining or improving its performance?

This paper presents our experiments on adapting the pretrained BERT-base model to the biomedical domain. We get the PubMed January 2024 baseline corpus and define different subset configurations using journal impact metrics: h-index (Hirsch, 2005) and Scimago Journal Rank or SJR (Guerrero-Bote and Moya-Anegón, 2012). We then perform continual pre-training from the BERT-base model (Devlin et al., 2019) and evaluate it on several tasks from the BLURB benchmark (Gu et al., 2022).

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 363–369
August 16, 2024. ©2024 Association for Computational Linguistics

2 Related work

2.1 Domain-adaptive and domain-specific pre-training for the biomedical domain

The adaptation of neural models to the biomedical domain has been extensively studied in recent years, focusing on BERT-type models and, more recently, large generative language models. We distinguish two main categories regarding the pre-training data:

- *Mixed-domain pre-training*, where the model has seen data from different domains during the pre-training: it can either be a model that has been pre-trained on a general corpus and then trained on in-domain data or a model trained simultaneously on data from multiple domains, such as biomedical and clinical for example (Lee et al., 2019; Beltagy et al., 2019; Peng et al., 2019).

- *Domain-specific pre-training*, where the model only sees data from a single domain during pre-training. The hypotheses are that by using a domain-specific vocabulary, the models learn more accurate representations of specific in-domain terms (that would be divided by the sub-word tokenization with a general corpus) and that it reduces noise introduced by text completely unrelated to the domain (Beltagy et al., 2019; Boukkouri et al., 2022; Lewis et al., 2020; Gu et al., 2022).

2.2 Pre-training data quality for large language models

Several works focus on selecting sequences using quality metrics for pre-training Transformer models in the general domain, particularly with the advent of large language models and the evolution of the size of pre-training datasets for these models (Zhou et al., 2023; Attendu and Corbeil, 2023; Marion et al., 2023; Das and Khetan, 2023).

The adaptation of large language models using scientific articles has been largely studied. However, only a few have emphasized the quality of scientific articles used. For the Galactica model (Taylor et al., 2022), they only mention applying *"several quality filters, including excluding papers from journals with certain keywords and also excluding papers with a low journal impact factor"*. Most other models that used PubMed or PubMedCentral for pre-training do not mention any specific selection of data at the document level; most focus on preprocessing steps at the content level (bibliography references, authors, figures and tables, etc.) when dealing with full-text articles (Luo et al., 2022; Wu et al., 2023; Luo et al., 2023; Chen et al., 2023).

3 Methods

We use the same methodology as Marion et al. (2023), with some small modifications :

Let D be a large dataset containing documents and ξ a metric assigning a score to a document. We build a subset $P_{c\xi}$ by adding instances that fit our selection criteria c :

$$P_{c\xi} = \{d_i \in D | c_{0\xi} \leq \xi(d_i)) \leq c_{1\xi}\} \qquad (1)$$

Where $c_{0\xi}$ and $c_{1\xi}$ are the lower and upper bound for the criteria c and the metric ξ. For each metric, we consider two selection criteria: keeping top or middle percentiles[1] of D as the data to be kept. This serves as verifying if the model learns better with high quality documents (defined by the metric, for our metrics, higher is better). We keep either 25% or 50% of the documents in D. So for instance, if we take the 25 % middle for the metric ξ, we should compute the 37.5 % and 62.5 % percentiles with respect to metric ξ, which corresponds to $c_{0\xi}$ and $c_{1\xi}$, and keep the documents between these two percentiles.

Then, we tokenize each document in the subset, and we concatenate them into sequences of length equal to the model's context length. This differs from Marion et al. (2023) as we do the filtering before tokenization (because our metrics are applied on a document, not on a sequence of tokens). These sequences are then used to pre-train a model. The goal is then to pre-train a model on a subset of the whole training set while retaining or improving the model's performance.

3.1 Pre-training corpus

We use the PubMed Baseline corpus comprising all article abstracts deposited on the PubMed database until January 2024. Using PubMed metadata, we filter out abstracts that are not in English, abstracts whose text is not available, and abstracts whose ISSN journal identifier is not present (we filter this to have enough abstracts with a score as our pruning metrics are based on journal impact). After

[1] we do not use the bottom percentiles because in our case, for the SJR metric, more than 25% of the dataset had the same value : 0

filtering, the total corpus is comprised of 15.9B tokens.

We did not perform a pre-training experiment using the non-filtered PubMed set because we did not have enough articles with journal identifiers to obtain convenient metric percentiles. Still, we expect this filtering to already impact the overall quality of the corpus.

3.2 Quality metrics

The nature of the datasets used for general model training (by which we mean models that are not domain-specific) differs from those used in the biomedical field. They are generally huge datasets comprising texts extracted from the Internet on various sites. In our case, these are research articles from the same database. This presupposes a text quality that is adequate in certain respects (generally correct syntax and formal language, unlike texts found on the Internet).

We wanted to use metrics specific to scientific articles that have meaning for scientific article readers. So, we decided to use journal impact metrics. We used the metadata available on PubMed. This type of metric can provide insight into the probable impact that a paper can have but does not necessarily ensure scientific quality. However, we believe filtering with impact metrics in a large corpus can help reduce the noise, help the model learn biomedical language, and learn biomedical knowledge more efficiently. We use the h-index (Hirsch, 2005) and the SJR (Guerrero-Bote and Moya-Anegón, 2012) as the data is publicly available on the Scimago website[2]. For comparison, we also perform a random score assignation on all papers from the dataset; we do not perform multiple random assignations to limit the compute cost.

We computed the percentiles for SJR and h-index and, as there were zero values for the SJR index (for the 12.5% and 25% percentiles), we did not perform all the pre-trainings for the *mid* criteria, we only considered the *25 %* subset. This is also why we did not consider the bottom percentiles. We also did not perform the pre-training on the complete set because of time and resource constraints, but we plan to do it in future work.

3.3 Pre-processing

We define

We tokenize the whole dataset and concatenate the text of the different abstracts into sequences of length 512 tokens (maximum sequence length for the model we use: BERT (Devlin et al., 2019)). . We keep 5 % of this set as validation data.

3.4 Model and pre-training

We use the original *BERT-base* model (Devlin et al., 2019), continue pre-training on the defined datasets with masked language modeling, and compare the resulting models. For each pre-training (on each subset), we fix a shared global number of steps so that each model sees the same quantity of tokens: we select the number of steps as the total number needed for one epoch on the entire PubMed corpus. For the runs with the subsets, the model will run multiple epochs until it reaches the total number of steps, with data shuffling between epochs (for example, two epochs for the run where we take the top 50% of PubMed abstracts with respect to h-index).

We train with a sequence length of 512 and a batch size of 8192[3], which gives us a total of 3598 steps. We use a linear schedule with 10 % warmup and a peak learning rate of $1e-4$. For the other hyperparameters, we follow the original BERT paper. We train our different models on 2 NVIDIA A100 GPUs.

3.5 Evaluation and fine-tuning

We evaluate the produced pre-trained models on some of the datasets from the BLURB benchmark (Gu et al., 2022). We also re-evaluate the BERT-based model to ensure a consistent evaluation with our fine-tuning scripts. We excluded the PICO and Sentence Similarity tasks (EBM-PICO (Nye et al., 2018) and BIOSSES (Soğancıoğlu et al., 2017)), for which we had trouble reproducing similar and consistent results across runs to those obtained in the BLURB paper, as they did not share any code to perform the fine-tuning and evaluation. So, we are left with the following evaluation tasks :

- Named entity recognition (NER) : BC5-chem & BC5-disease (Li et al., 2016), BC2GM (Smith et al., 2008), JNLPBA (Collier and Kim, 2004) and NCBI-disease (Doğan et al., 2014). We evaluate the models for NER tasks using the *entity-level F1 score*. We model the entities using BIO tags.

[2] https://www.scimagojr.com/journalrank.php

[3] We perform gradient accumulation and data parallelism to get this batch size.

	base	random		h-index				sjr		
				mid		top		mid	top	
	0%	25%	50%	25%	50%	25%	50%	25%	25%	50%
BC5-chem	87.31	90.03	**90.24**	89.40	89.93	89.51	89.52	89.72	89.61	89.89
BC5-disease	77.09	**81.09**	80.72	81.05	80.73	80.38	80.68	81.00	80.76	80.60
BC2GM	75.32	79.17	79.01	79.51	79.51	79.52	79.41	78.74	79.01	**79.87**
JNLPBA	76.77	78.02	77.85	77.51	77.95	78.13	**78.41**	78.13	78.28	77.90
NCBI-disease	81.59	84.89	84.45	**85.09**	84.84	84.63	84.71	84.97	84.30	84.98
HoC	79.22	84.41	84.74	84.72	84.56	84.83	84.71	84.54	**85.07**	84.76
ChemProt	77.07	79.25	78.83	78.94	79.72	79.00	**79.92**	78.77	79.62	78.96
DDI	**89.11**	87.54	87.70	87.91	88.27	86.46	86.80	87.05	85.92	87.76
GAD	76.82	78.09	78.24	78.31	77.38	77.34	**78.39**	77.42	78.35	77.00
BioASQ	72.19	75.93	75.63	75.63	75.24	74.84	**76.07**	75.85	75.50	75.22
PubMed QA	**55.24**	55.20	55.20	55.16	55.12	54.78	55.16	55.20	55.22	55.20
Micro avg.	77.07	79.42	79.33	79.38	79.39	79.04	**79.43**	79.22	79.24	79.29
Macro avg.	75.89	78.56	78.55	78.59	78.53	78.25	**78.64**	78.41	78.53	78.46

Table 1: Comparison of the performance of our pretrained models on the different evaluation tasks from the BLURB benchmark (Gu et al., 2022). *'base'* model is the BERT$_{BASE}$ model (Devlin et al., 2019) from which we continue the pre-training. For the macro average, we average the datasets from the same task and then average the performance on each task. For each task or average, the **best performance is in bold** and the second best performance is underlined.

- Relation extraction : ChemProt(M. et al., 2017), DDI (Herrero-Zazo et al., 2013), GAD (Bravo et al., 2015). We evaluate the models for relation extraction using the *micro F1 score*. We use entity dummyfication with start and end tags and use the [CLS] token to classify relations.

- Document classification : HoC (Baker et al., 2016), for which we measure the *micro F1 score*.

- Question answering : PubMedQA (Jin et al., 2019) and BioASQ Task 7b (Nentidis et al., 2020). We evaluate these tasks using *accuracy*.

4 Results and Discussion

To limit random effects, we perform the fine-tuning multiple times with different random seeds, as described in the BLURB paper: using five seeds for all datasets except for BioASQ and PubMedQA, for which we use ten seeds (because they are smaller in size). We then report the average performance across the different seeds for each dataset in the table 1.

4.1 Improvement against non biomedical model

All models trained on biomedical data perform better than the base model trained only on general-domain data. However, for a fair comparison, we should train it for the same amount of steps on non-biomedical data.

4.2 Are journal impact metrics important for the model ?

We obtain the best results in micro and macro averages for the model trained on the top 50% of the entire set with respect to the h-index of the journal in which abstracts have been published. Overall, the h-index metric performs better than SJR, which may be because the SJR percentile values are very close to each other, so the quality differences are less important.

However, the performance differences are low when we compare to the SJR metric or even when selecting abstracts randomly, regardless of the proportion of abstracts we keep. So, journal impact metrics do not seem important when selecting pretraining data from a corpus of scientific articles. We then should find more appropriate metrics to define the quality of a single abstract or test it on a full-text article corpus (so that the impact of a single document is higher).

4.3 Is it better to pre-train a model using more abstracts ?

If we compare the performance difference when training with 25% of the data against 50%, we globally have better performances (except for the random selection), but these differences are not significant. So, it would be interesting to perform further pre-training experiments using different subset sizes to investigate which number of documents is optimal for the domain adaptation.

5 Conclusion

This paper presents our early experiments on selecting the pre-training data for the biomedical domain. We show that the journal impact metrics are not better than the random selection at a fixed number of training steps. We also observe that reducing the number of abstracts in the training set does not necessarily decrease the final model performance and show the need to investigate how many documents we need to pre-train a model without losing performance.

Further directions include finding better metrics (or combinations of metrics) to assess the quality of a document in the pre-training corpus, investigating metrics at a different level (at the corpus level using various mixtures of biomedical domains), and using a corpus of full-text articles.

6 Acknowledgments

This project was provided with computer and storage resources by GENCI at IDRIS thanks to the grant 20XX-AD011014707 on the supercomputer Jean-Zay's A100 partition .

References

Jean-michel Attendu and Jean-philippe Corbeil. 2023. NLU on Data Diets: Dynamic Data Subset Selection for NLP Classification Tasks. In *Proceedings of The Fourth Workshop on Simple and Efficient Natural Language Processing (SustaiNLP)*, pages 129–146, Toronto, Canada (Hybrid). Association for Computational Linguistics.

Simon Baker, Ilona Silins, Yufan Guo, Imran Ali, Johan Högberg, Ulla Stenius, and Anna Korhonen. 2016. Automatic semantic classification of scientific literature according to the hallmarks of cancer. *Bioinformatics*, 32(3):432–440.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Hicham El Boukkouri, Olivier Ferret, Thomas Lavergne, and Pierre Zweigenbaum. 2022. Re-train or train from scratch? Comparing pre-training strategies of BERT in the medical domain. In *LREC 2022 - Language Resources and Evaluation Conference*, page 2626.

Àlex Bravo, Janet Piñero, Núria Queralt-Rosinach, Michael Rautschka, and Laura I. Furlong. 2015. Extraction of relations between genes and diseases from text and large-scale data analysis: Implications for translational research. *BMC Bioinformatics*, 16(1):55.

Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, Alexandre Sallinen, Alireza Sakhaeirad, Vinitra Swamy, Igor Krawczuk, Deniz Bayazit, Axel Marmet, Syrielle Montariol, Mary-Anne Hartley, Martin Jaggi, and Antoine Bosselut. 2023. MEDITRON-70B: Scaling Medical Pretraining for Large Language Models. *arXiv preprint*.

Nigel Collier and Jin-Dong Kim. 2004. Introduction to the Bio-entity Recognition Task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and Its Applications (NLPBA/BioNLP)*, pages 73–78, Geneva, Switzerland. COLING.

Devleena Das and Vivek Khetan. 2023. DEFT: Data Efficient Fine-Tuning for Large Language Models via Unsupervised Core-Set Selection. *arXiv preprint*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. NCBI disease corpus: A resource for disease name recognition and concept normalization. *Journal of Biomedical Informatics*, 47:1–10.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2022. Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23.

Vicente P. Guerrero-Bote and Félix Moya-Anegón. 2012. A further step forward in measuring journals' scientific prestige: The SJR2 indicator. *Journal of Informetrics*, 6(4):674–688.

María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. 2013. The DDI corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of Biomedical Informatics*, 46(5):914–920.

J. E. Hirsch. 2005. An index to quantify an individual's scientific research output. *Proceedings of the National Academy of Sciences of the United States of America*, 102(46):16569–16572.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A Dataset for Biomedical Research Question Answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, page btz682.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020. Pretrained Language Models for Biomedical and Clinical Tasks: Understanding and Extending the State-of-the-Art. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 146–157, Online. Association for Computational Linguistics.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. 2016. BioCreative V CDR task corpus: A resource for chemical disease relation extraction. *Database: The Journal of Biological Databases and Curation*, 2016:baw068.

Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Shenmin Zhang, Hoifung Poon, and Tie-Yan Liu. 2022. BioGPT: Generative Pre-trained Transformer for Biomedical Text Generation and Mining. *undefined*.

Yizhen Luo, Jiahuan Zhang, Siqi Fan, Kai Yang, Yushuai Wu, Mu Qiao, and Zaiqing Nie. 2023. BioMedGPT: Open Multimodal Generative Pretrained Transformer for BioMedicine. *arXiv preprint*.

Krallinger M., Rabal O., and Lourenço A. 2017. Overview of the biocreative vi chemical-protein interaction track. *Proceedings of the BioCreative VI Workshop,*, 141-146.

Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. 2023. When Less is More: Investigating Data Pruning for Pretraining LLMs at Scale. *arXiv preprint*.

Anastasios Nentidis, Konstantinos Bougiatiotis, Anastasia Krithara, and Georgios Paliouras. 2020. Results of the seventh edition of the BioASQ Challenge. volume 1168, pages 553–568.

Benjamin Nye, Junyi Jessy Li, Roma Patel, Yinfei Yang, Iain Marshall, Ani Nenkova, and Byron Wallace. 2018. A Corpus with Multi-Level Annotations of Patients, Interventions and Outcomes to Support Language Processing for Medical Literature. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 197–207, Melbourne, Australia. Association for Computational Linguistics.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer Learning in Biomedical Natural Language Processing: An Evaluation of BERT and ELMo on Ten Benchmarking Datasets.

Sebastian Ruder, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf. 2019. Transfer Learning in Natural Language Processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 15–18, Minneapolis, Minnesota. Association for Computational Linguistics.

Larry Smith, Lorraine K. Tanabe, Rie Johnson nee Ando, Cheng-Ju Kuo, I-Fang Chung, Chun-Nan Hsu, Yu-Shi Lin, Roman Klinger, Christoph M. Friedrich, Kuzman Ganchev, Manabu Torii, Hongfang Liu, Barry Haddow, Craig A. Struble, Richard J. Povinelli, Andreas Vlachos, William A. Baumgartner, Lawrence Hunter, Bob Carpenter, Richard Tzong-Han Tsai, Hong-Jie Dai, Feng Liu, Yifei Chen, Chengjie Sun, Sophia Katrenko, Pieter Adriaans, Christian Blaschke, Rafael Torres, Mariana Neves, Preslav Nakov, Anna Divoli, Manuel Maña-López, Jacinto Mata, and W. John Wilbur. 2008. Overview of BioCreative II gene mention recognition. *Genome Biology*, 9(2):S2.

Gizem Soğancıoğlu, Hakime Öztürk, and Arzucan Özgür. 2017. BIOSSES: A semantic sentence similarity estimation system for the biomedical domain. *Bioinformatics*, 33(14):i49–i58.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. 2022. Galactica: A Large Language Model for Science. *arXiv preprint*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Chaoyi Wu, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2023. PMC-LLaMA: Further Finetuning LLaMA on Medical Papers. *arXiv preprint*.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. LIMA: Less Is More for Alignment. *arXiv preprint*.

Leveraging LLMs and Web-based Visualizations for Profiling Bacterial Host Organisms and Genetic Toolboxes

Gilchan Park[1*], Vivek Mutalik[2], Christopher Neely[2], Carlos Soto[1], Shinjae Yoo[1], Paramvir Dehal[2]

[1]Brookhaven National Laboratory, Upton, New York, USA
[2]Lawrence Berkeley National Laboratory, Berkeley, California, USA
***Correspondence:** gpark@bnl.gov

Abstract

Building genetic tools to engineer microorganisms is at the core of understanding and redesigning natural biological systems for useful purposes. Every project to build such a genetic toolbox for an organism starts with a survey of available tools. Despite a decade-long investment and advancement in the field, it is still challenging to mine information about a genetic tool published in the literature and connect that information to microbial genomics and other microbial databases. This information gap not only limits our ability to identify and adopt available tools to a new chassis but also conceals available opportunities to engineer a new microbial host. Recent advances in natural language processing (NLP), particularly large language models (LLMs), offer solutions by enabling efficient extraction of genetic terms and biological entities from a vast array of publications. This work present a method to automate this process, using text-mining to refine models with data from bioRxiv and other databases. We evaluated various LLMs to investigate their ability to recognize bacterial host organisms and genetic toolboxes for engineering. We demonstrate our methodology with a web application that integrates a conversational LLM and visualization tool, connecting user inquiries to genetic resources and literature findings, thereby saving researchers time, money and effort in their laboratory work. The code and data are available at: https://github.com/boxorange/LLM-GeneticTool-Extraction

1 Introduction

Our planet currently faces significant challenges concerning biological resources, including limited renewable energy sources, lack of innovative treatments for endemic infectious diseases, water pollution, insufficient arable land resulting in food crises, and the degradation of ecosystems (WEF, 2020; Arkin et al., 2010), among other urgent issues. We postulate – as others have – that the capacity to domesticate and genetically engineer non-model microorganisms from relevant environments could facilitate the development of potential solutions to many of these urgent global problems (Endy, 2005; Stacey, 2017). Although recent technological advances have been made at a rapid pace to address several of these challenges, the information needed for each potential new model organism is dispersed across the literature and is not readily accessible to many practitioners. This situation complicates every new synthetic biology, bioenergy, and biomanufacturing project involving a non-model organism (Mutalik et al., 2013; Council et al., 2015). The disorganized nature of the information not only impedes machine-readable approaches but also hinders the assessment of the scope of work and identification of knowledge gaps, subsequently offering limited guidance for investment to overcome technological barriers. For instance, despite decades of progress in the field of synthetic biology, it remains challenging to pinpoint suitable microbial targets for specific applications and conditions, as well as the genetic tools required for cultivating and engineering non-model microorganisms (Arkin, 2008; Council et al., 2015; Oberhardt et al., 2015; Price and Arkin, 2017).

A comprehensive literature mining tool that monitors emerging technologies and genetic tools critical for biotechnology professionals would be highly beneficial. This envisioned tool would allow us to identify information gaps and detect opportunities concealed within extensive literature. For instance, the tool should efficiently ascertain whether a chosen organism is suitable for laboratory domestication and which genetic tools are available for that organism, streamlining the search process and conserving time, effort, and funding for numerous lab-oriented projects.

Recent advancements in Natural Language Pro-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 370–379
August 16, 2024. ©2024 Association for Computational Linguistics

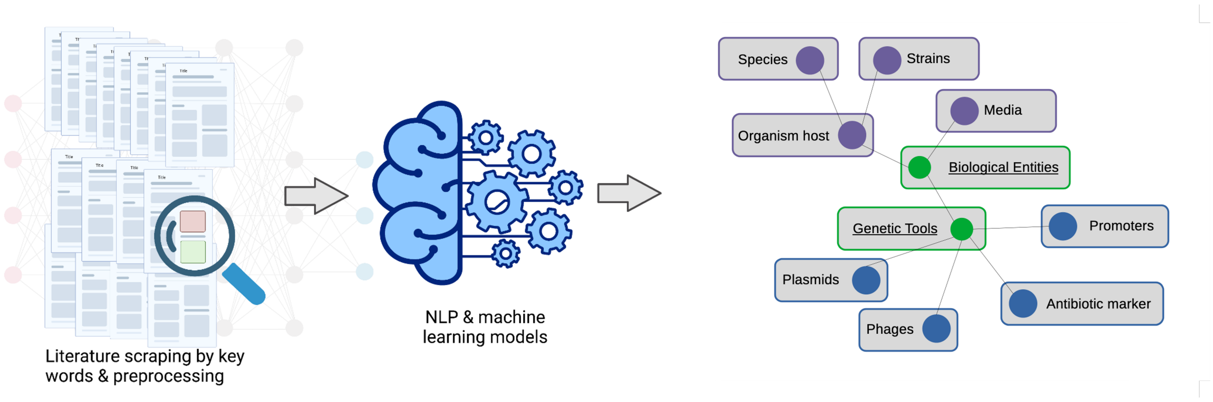


Figure 1: The project objective and workflow: We propose to use NLP and machine learning models to process and learn from literature data about growth characteristics, conditions, traits and available *in silico* models and genetic tools to engineer microorganisms.

cessing (NLP) and Large Language Model (LLM) have enabled the analysis of textual data at an unprecedented scale (e.g., millions of documents), allowing for the extraction of significant contextual information in ways that were previously unachievable. These innovative techniques offer considerable potential to bridge the knowledge gap discussed earlier. In this study, we propose employing NLP on biological literature to identify organism traits (as depicted in Figure 1) and systematically compiling this extracted knowledge. This approach facilitates the development of automated, curated centralized systems crucial for the cultivation and engineering of microorganisms.

This work specifically focuses on the extraction of information about bacterial organism hosts and genetic toolbox for them from literature. As detailed in this paper, our contribution is threefold:

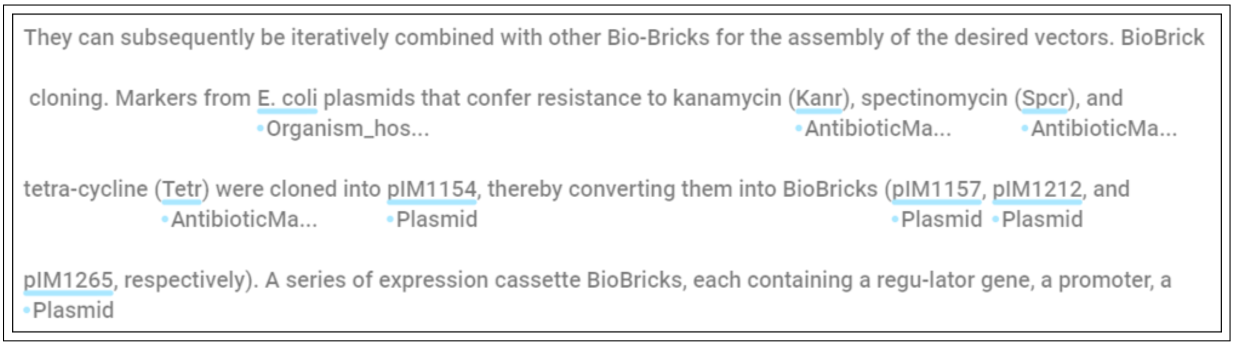
1. We present a curated corpus of bacterial host organisms and genetic toolboxes, classified into 14 distinct labels derived from scientific literature, including plasmids, promoters, reporters, and other entities of interest. The selection of a bacterial host organism is determined by the accessibility and suitability of genetic toolbox for efficient manipulation and engineering. This, in turn, informs the feasibility and ease of engineering the selected host organism for targeted applications within the synthetic biology field.

2. This paper provides an evaluation of various publicly available LLMs for the task of recognizing organism hosts and genetic tools. Our findings demonstrate the efficacy of fine-tuning LLMs on an annotated dataset, which enhances the models' performance in those entity recognition.

3. This work presents a chatbot interface designed to facilitate discussions between users and a specialized chatbot. The chatbot leverages public biological resource such as NCBI taxonomy, genetic tool databases, and publication information. Users can pose questions regarding the genetic engineerability of biological entities and engage in informative dialogues on the subject.

2 Related Work

LLMs demonstrated significant improvements in addressing a multitude of NLP tasks that are critical to the fields of biology and biomedicine (Chen et al., 2023; Yu et al., 2024). Instructed on a broad spectrum of text corpora, encompassing web crawls, medical records, and rigorously selected datasets, LLMs are equipped with the proficiency to integrate information from diverse sources. These sources range from scientific publications and databases to various other forms of informational repositories. This integrative ability enables LLMs to identify complex interconnections, nuanced contextual aspects, and insights that may remain obscure to traditional methods. The BioMistral model (Labrak et al., 2024), built upon the Mistral foundational model and subsequently pre-trained on PubMed Central, underwent evaluation on a benchmark encompassing 10 medical question-answering (QA) tasks. This assessment revealed superior performance compared to the original model and existing open-source medical counterparts.



They can subsequently be iteratively combined with other Bio-Bricks for the assembly of the desired vectors. BioBrick

cloning. Markers from E. coli plasmids that confer resistance to kanamycin (Kanr), spectinomycin (Spcr), and

tetra-cycline (Tetr) were cloned into pIM1154, thereby converting them into BioBricks (pIM1157, pIM1212, and

pIM1265, respectively). A series of expression cassette BioBricks, each containing a regu-lator gene, a promoter, a

Figure 2: An example of annotation through the Doccano web server.

McInnes et al. (2022) presents the development of a synthetic biology knowledge system, wherein a text processing pipeline utilizes NLP techniques to extract and correlate information from the literature aimed at synthetic biology researchers. The pipeline integrates named entity recognition, relation extraction, concept grounding, and topic modeling methodologies to extract pertinent information from published literature. Subsequently, this extracted information is utilized to establish connections between articles and elements within the knowledge system. The findings demonstrate the effectiveness of each component when applied to synthetic biology literature and propose avenues for further enhancing the pipeline's capabilities. Gong et al. (2023) explored the potential of various LLMs such as GPT-4, GPT-3.5, PaLM2, Claude2, and SenseNova in addressing conceptual biology questions, including those related to synthetic biology, such as principles of genetic circuit design and CRISPR-based genome editing techniques. While the findings revealed the adeptness of LLMs in logical reasoning and their potential to support biology research by facilitating tasks such as data analysis, hypothesis formulation, and knowledge synthesis, the authors underscored the necessity for further refinement and validation before fully harnessing the potential of LLMs to expedite biological discovery.

In this study, we assessed the effectiveness of LLMs in recognizing and extracting pertinent information regarding host organisms and their associated genetic engineering tools. Of particular emphasis was the evaluation of open-source LLMs, chosen for their heightened adaptability and transparency in contrast to proprietary counterparts, thereby enabling users to exercise greater customization and oversight over model operations. The main objective was to gauge the efficacy of these models in discerning insights from a collection of biological literature and resources, thereby augmenting our comprehension of LLMs' applicability in biological inquiry and ability to inform prospective applications within this domain.

3 Host Organisms and Genetic Toolbox Curation

To the best of our knowledge, there are no publicly available datasets specifically tailored for the recognition task involving bacterial host organisms and their associated genetic toolbox by machine learning models. In order to enhance the proficiency of machine learning models in identifying such entities through training on labeled datasets, we undertook an annotation endeavor aimed at labeling both organism hosts and genetic tool types as depicted in biological literature. To facilitate this annotation task, a comprehensive list of terms describing the bacterial genetic toolbox was curated, which is provided in Appendix A. When conducting a search using the genetic toolbox related terms, numerous papers unrelated to our specific focus emerge. These papers span topics ranging from human genetics to plant research. While we may consider including them at a later stage, our current emphasis lies on bacteria. Therefore, we incorporate the compound word "bacteria" along with relevant keywords to refine our search results.

For the annotation process, a corpus comprising 434 PDF papers was assembled, meticulously selected by two domain experts. Additionally, 376 XML articles containing any of the terms from the curated list described in Appendix A were obtained by querying the bioRxiv database within subject areas encompassing *Biochemistry, Bioengineering, Bioinformatics, Microbiology, Molecular Biology, Synthetic Biology, and Systems Biology*. From this corpus, a total of 795 text snippets were extracted from abstracts and main body texts, each compris-

Entity	The number of entities
Plasmid	445
Organism Host	407
Promoter	181
Genome Engineering	169
Cloning Method	158
Reporter	122
Regulator	86
Antibiotic Marker	65
Genetic Screen	40
RBS†	35
Counter Selection	27
Terminator	23
DNA Transfer	14
Operator	5

Table 1: The statistics of 1,777 annotated labels for organism hosts and genetic tools. †RBS stands for Ribosome Binding Site.

ing a target sentence containing one of the bacterial genetic toolbox terms, accompanied by two preceding and two succeeding sentences, all of which are part of the same paragraph. To ensure non-redundant annotation, we eliminated duplicate snippets that contain multiple genetic toolbox terms.

To facilitate the annotation process, a Doccano web server (Nakayama et al., 2018) was employed, thereby streamlining the task of annotating textual data. An annotation sample is depicted in Figure 2. In order to label entities within the text, a framework comprising 14 distinct entity labels was defined. Subsequently, a total of 1,777 annotated entities were obtained across the entire corpus following the completion of the annotation process. Table 1 presents the 14 labels along with the corresponding number of entities.

4 Evaluation of LLMs for Recognizing Host Organisms and Genetic Tool Types

Our study aimed to assess the potential of LLMs for the task of entity type recognition, utilizing annotated datasets. To this end, we employed a selection of LLMs, namely Falcon (Almazrouei et al., 2023), MPT (MosaicML-NLP-Team, 2023), LLaMA 2 (Touvron et al., 2023), SOLAR (Kim et al., 2023), Mistral (Jiang et al., 2023), Mixtral (Jiang et al., 2024), and LLaMA 3 (Meta-AI, 2024). Model evaluation was conducted utilizing a question answering formatted prompt paired with text snippets acquired from our annotation task. An illustrative sample of such a prompt for the entity type recognition task is provided below.

> **Question:** Given the options: “plasmid”, “organism host”, “promoter”, “genome engineering”, “cloning method”, “reporter”, “regulator”, “antibiotic marker”, “genetic screen”, “RBS”, “counter selection”, “terminator”, “DNA transfer”, “operator”, which one is the entity type of J23108 in this text?
>
> **Text:** Plasmids were cloned using Gibson Assembly or inverse PCR, propagated in E. coli TG1 competent cells in LB media, and isolated through miniprep (Qiagen.) Reporter plasmids had a p15A origin of replication, chloramphenicol resistance, and the terminator trrnB downstream of the sfGFP coding sequence. Plasmids for overexpressing ribosomal proteins in vivo had a ColE1 origin of replication, ampicillin resistance, the synthetic constitutive E. coli promoter J23108 from the Registry of Standard Biological Parts, and the terminator trnnB after the protein expression gene.
>
> **Answer:** promoter

The 1,777 text snippets underwent partitioning into distinct train, validation, and test sets, maintaining an 8:1:1 ratio. A comparative analysis was then conducted between the original pre-trained models and their fine-tuned counterparts. To facilitate the fine-tuning process, we employed the Low-Rank Adaptation (LoRA) technique (Hu et al., 2021) coupled with quantization (QLoRA) (Dettmers et al., 2024), aiming to enhance memory efficiency and expedite training procedures. The training was performed on all linear layers within the models. The experiments were conducted on 4×NVIDIA A100 80GB GPUs. The configurations for fine-tuning the models were established as follows.

- **Batch size**: 2
- **Training epochs**: 5
- **QLoRA target modules**: all linear layers
- **Quantization technique**: BitsandBytes
- **Quantization**: 4-bit
- **Learning rate**: 1e-4 with AdamW

Table 2 presents the micro and macro F1-scores derived from the evaluation of the original pre-trained LLMs and those fine-tuned with QLoRA adaptation for the entity type recognition task employing zero-shot prompting. The results indicate that the LLaMA 3 (70B) model demonstrated superior prediction capability compared to other original LLMs. Notably, following adaptation to the

Model	Context Length	Original		QLoRA adapted	
		Micro F1	Macro F1	Micro F1	Macro F1
Falcon (7B)	2K	0.4213	0.1795	0.8933	0.7672
Falcon (40B)	2K	0.6966	0.4106	0.8820	0.6857
MPT-Chat (7B)	2K	0.5618	0.3814	0.8708	0.6503
MPT-Chat (30B)	8K	0.7697	0.6152	0.9213	0.8160
LLaMA-2-Chat (7B)	4K	0.5562	0.3703	0.8933	0.6810
LLaMA-2-Chat (70B)	4K	0.7584	0.5701	0.9157	0.8087
SOLAR-10.7B-Instruct (30B)	4K	0.7528	0.5815	0.9045	0.7252
Mistral-7B-Instruct (7B)	8K	0.7022	0.5565	0.9326	0.8351
Mixtral-8x7B-Instruct (46B)	32K	0.7135	0.5499	**0.9607**	0.7585
LLaMA-3 (8B)	8K	0.6573	0.4969	0.9270	0.6867
LLaMA-3 (70B)	8K	**0.8708**	**0.6194**	0.9551	**0.8557**

Table 2: The original pre-trained and QLoRA fine-tuned LLMs evaluation on the entity type recognition task with zero-shot prompting.

QLoRA framework, the performance of the LLM models exhibited a substantial improvement, with the LLaMA 3 (70B) and Mixtral 8x7B model as the top performer. The macro F1-scores, being lower than the micro F1-scores, suggest challenges encountered by the models in accurately identifying certain classes, such as “RBS” and “genetic screen”. The potential ambiguity or variability in interpretation of these terms may arise particularly in instances where the training data available to the models lacks comprehensive examples of the term’s utilization within the context of genetic techniques. Moreover, the effective application and interpretation of these terms can require specialized knowledge, which may be acquired through access to specific datasets within the respective field. To enhance the models’ capacity to discern and comprehend the intended meaning of such terms, it might be necessary to integrate supplementary sources of information, such as external knowledge repositories, and to provide explicit contextual explanations upon their introduction.

5 Preliminary Evaluation of a LLM for Extracting Host Organisms and Genetic Tools from Raw Articles

Our primary objective is to develop an automated tool capable of extracting host organisms and genetic tools for engineering purposes from unprocessed textual data. Such a tool would offer practitioners and domain experts a more streamlined and convenient means of accessing this information. As a preliminary phase, we initially evaluated the performance of a plain LLM in identifying biological entities and genetic tools within scientific literature. Employing a question-answering framework, we devised a structured prompt using the full text of each article. The following exemplifies a sample prompt for this specific task, incorporating the entire text of an article.

> **Question:** Which biological entities and genetic tools are mentioned in this paper, and what are their types? Choose from the following options: “plasmid”, “organism host”, “promoter”, “genome engineering”, “cloning method”, “reporter”, “regulator”, “antibiotic marker”, “genetic screen”, “RBS”, “counter selection”, “terminator”, “DNA transfer”, “operator”
>
> **Paper:** Title: A BioBricks® toolbox for multiplexed metabolic engineering of central carbon metabolism in the tetracenomycin pathway. Full-Text: The tetracenomycins are aromatic anticancer polyketides that inhibit peptide translation via binding to the large ribosomal subunit. Here, we expressed the elloramycin biosynthetic gene cluster ...
>
> **Answer:** The biological entities and genetic tools mentioned in this paper are: 1. Organism host: "Streptomyces coelicolor M1146, Streptomyces glaucescens GLA.0, Streptomyces olivaceus TTü2353, Amycolatopsis sp. A23, E. coli JM109, E. coli ET12567/pUZ8002 2. Promoter: ermE*p promoter, sf14p promoter, LacI promoter 3. Genome engineering: cos16F4iE cluster, actinophage integrases (e.g. fC31, fBT1, SV1, TG1, SAM2, VWB) 4. Cloning method: restriction digestion, ligation, 3A cloning, isocaudomer cloning 5. Reporter: red fluorescent protein (RFP) coding device 6. Regulator: LacI repressor 7. Antibiotic marker: ampicillin, kanamycin, apramycin, viomycin, hygromycin, nalidixic acid 8. Genetic screen: blue-white colony screening 9. RBS: B0034 strong ribosome binding site 10. Counter selection: nalidixic acid 11. Terminator: B0015 transcriptional terminator ....

As a preliminary evaluation, we employed 233 articles used in our annotation task. Due to the length of a full text, only the Mixtral 8x7B model was adopted for this evaluation, which can have up to 32K input length. The post-processing of the model output generated a JSON formatted data. The following illustrates an example of post-processed model prediction.

```
Title: "A BioBricks toolbox for multiplexed metabolic
        engineering of central carbon metabolism
        in the tetracenomycin pathway"
{
    "antibiotic marker": [
        "ampicillin, kanamycin, apramycin, viomycin,
        hygromycin, nalidixic acid"
    ],
    "cloning method": [
        "restriction digestion, ligation, 3A cloning,
        isocaudomer cloning"
    ],
    "counter selection": [
        "nalidixic acid"
    ],
    "dna transfer": [
        "intergeneric conjugation, conjugal matings,
        electroporation, ..."
    ],
    "genetic screen": [
        "blue-white colony screening"
    ],
    "genome engineering": [
        "cos16F4iE cluster, actinophage integrases
        (e.g. fC31, fBT1, SV1, TG1, SAM2, VWB)"
    ],
    "operator": [
        "attB site, attP site, oriT, attP site"
    ],
    "organism host": [
        "Streptomyces coelicolor M1146,
        Streptomyces glaucescens GLA.0, ..."
    ],
    "promoter": [
        "ermE*p promoter, sf14p promoter, LacI promoter"
    ],
    "RBS": [
        "B0034 strong ribosome binding site"
    ],
    "regulator": [
        "LacI repressor"
    ],
    "reporter": [
        "red fluorescent protein (RFP) coding device"
    ],
    "terminator": [
        "B0015 transcriptional terminator"
    ]
}
```

Two domain experts vetted this model prediction for species and tool names/types detection, and the model's performance is displayed in Table 3. The result shows 0.8816 (micro F1) and 0.8734 (macro F1) for 6,962 entities. The model displays inherent uncertainty when encountering ambiguous terminology. For example, the term "genetic screen" has been utilized across diverse contexts, resulting in confusion within the model. This assertion is supported by the individual accuracy measurements presented in Table 4, where "genetic screen" exhibited the lowest level of precision. A similar observation was made in an earlier experiment, during which the models encountered difficulties in recognizing "genetic screen".

Model	**Micro F1**	**Macro F1**
Mixtral-8x7B-Instruct	0.8816	0.8734

Table 3: The Mixtral 8x7B model's evaluation on the host organism and genetic toolbox extraction task with zero-shot prompting.

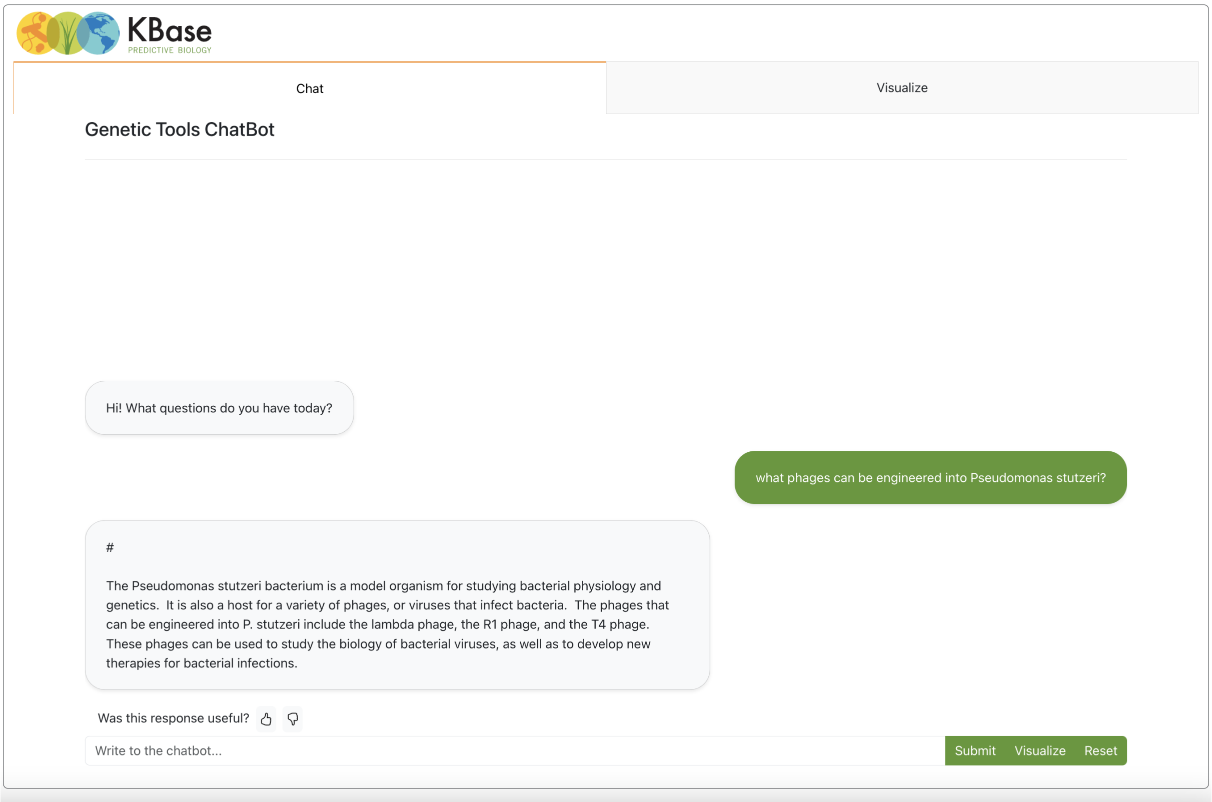
Entity	**Count**	**Accuracy**
Plasmid	1485	0.8936
Organism Host	716	0.8282
Promoter	656	0.8872
Genome Engineering	601	0.8602
Antibiotic Marker	525	0.9295
Regulator	501	0.9122
Cloning Method	498	0.8313
Reporter	434	0.8594
Operator	356	0.9719
DNA Transfer	356	0.9129
RBS	233	0.8670
Genetic Screen	221	0.8281
Terminator	197	0.8782
Counter Selection	183	0.8634

Table 4: Individual Entity Accuracy

6 Chatbot for Genetic Tool Engineering

Complementing the development of a LLM to assist research in synthetic biology and biomanufacturing, KBase (Arkin et al., 2018) provides a web application for users to interact with this model through a chatbot interface. Starting from a simple question, users can ask for information about bacteria and genetic tools of interest and receive responses from the trained model. Conversations are logged, allowing users to provide feedback on the efficacy of the chatbot's responses, which serves as valuable input for refining and enhancing the system in future iterations (see Figure 3).

Using the "outlines" Python package (Willard and Louf, 2023) for structured output generation, we identify any biological entities and their associated tools in the model's response. The genus of each entity is collected, and a pruned NCBI taxonomy tree (Schoch et al., 2020) is rendered that highlights these organisms in the context of their genus-level neighbors. As a result of the continuous evolution of taxonomic nomenclature, the information output by the model may not reflect the current information in NCBI databases. Therefore, this tool performs additional checks against previous names and synonyms for organisms iden-



Genetic Tools ChatBot

Hi! What questions do you have today?

what phages can be engineered into Pseudomonas stutzeri?

#

The Pseudomonas stutzeri bacterium is a model organism for studying bacterial physiology and genetics. It is also a host for a variety of phages, or viruses that infect bacteria. The phages that can be engineered into P. stutzeri include the lambda phage, the R1 phage, and the T4 phage. These phages can be used to study the biology of bacterial viruses, as well as to develop new therapies for bacterial infections.

Was this response useful?

Write to the chatbot...

Figure 3: A screen shot of the LLM-powered Chatbot in the web application.

tified in the model output. The web application downloads the most current version of the NCBI taxonomy database at container startup.

Users may select any organisms on the species tree, and the selection will be summarized on the right side of the visualizer. The top right provides summary information from the BacDive database (Reimer et al., 2022) to rapidly identify culture conditions amenable to isolation and growth. The bottom right section summarizes genetic tool database information from the Phage-Host Daily (Albrycht et al., 2022), Virus-Host (Mihara et al., 2016), and Plasmid (Schmartz et al., 2022) databases. Additional information identified by the model is also described here. This combination of prepackaged curated databases in conjunction with information extracted from the model's large text corpora provides a comprehensive summary of the available strains, tools, and publications describing the organism in question (see Figure 4).

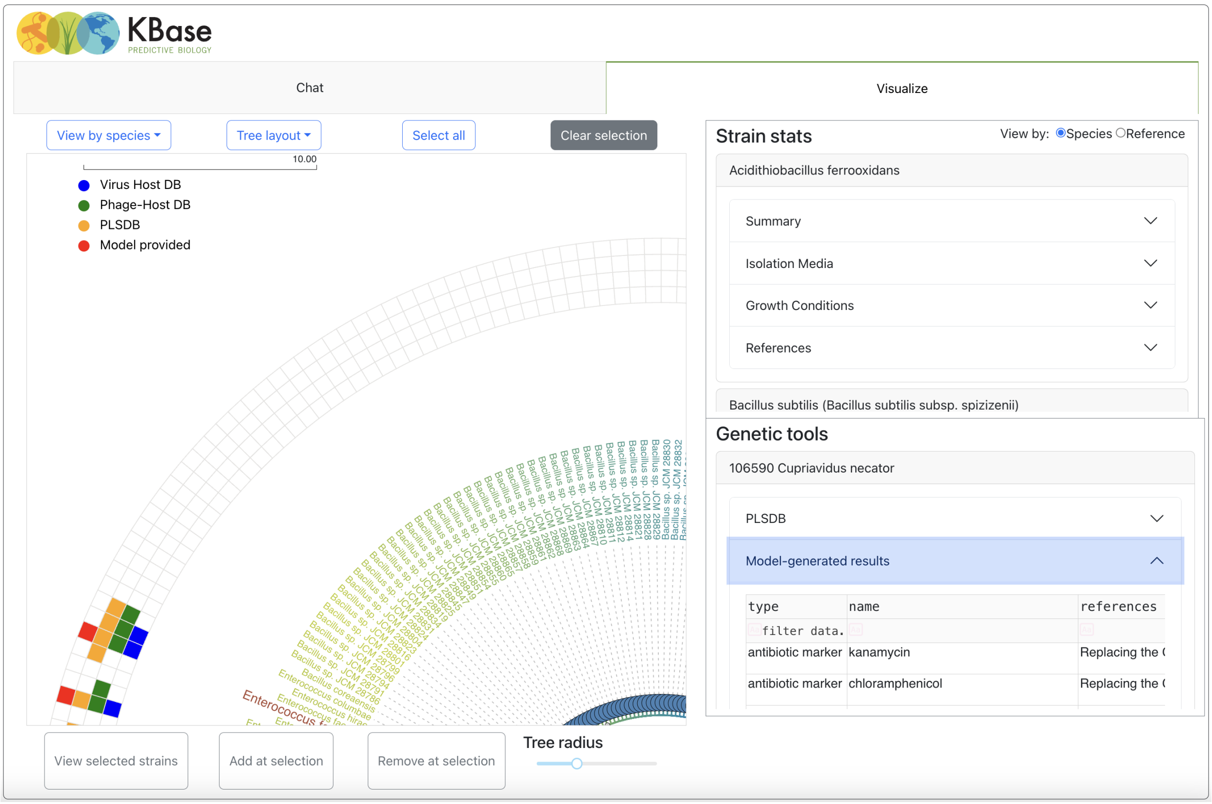
To facilitate the identification of organisms that do not have isolation or genetic tool information, users may elect to visualize other relative organisms without entries in the accompanying databases. Similarly, while the tree visualization is species-focused, users may also select to view all strains for a given species in order to highlight strain-level differences in isolation and genetic tool usage.

The incorporation of this tool into the KBase infrastructure serves as an additional avenue for researchers to access pertinent information, establishing connections not only with biologically relevant organisms for laboratory investigations but also with the broader ecosystem of KBase, facilitating subsequent analyses and dissemination of findings.

Integration with the KBase platform is undertaken in adherence to established standards pertaining to containerization, user-oriented tool development, and deployment protocols. The tool is being developed with the intent of serving as a reusable proof-of-concept that caters to a diverse audience.

7 Conclusion

A significant bottleneck within the domains of synthetic biology and biomanufacturing pertains to the identification of suitable microbial targets tailored to specific applications and environmental conditions, alongside the selection of genetic tools conducive to the cultivation and engineering of non-model microorganisms. This bottleneck poses



type	name	references
filter data.		
antibiotic marker	kanamycin	Replacing the (
antibiotic marker	chloramphenicol	Replacing the (

Figure 4: A screen shot of the species tree view in the web application.

a direct hindrance to the progress of investigations in synthetic biology and obstructs redesign efforts. While peer-reviewed publications serve as the primary repository for biological experimental data, manual curation proves insufficient for managing the extensive volume of available literature. Consequently, there arises a need for the automated extraction of data related to environmental conditions and genetic tools from literature sources. Recent advancements in NLP and LLMs have presented promising avenues for addressing this challenge. This study aims to evaluate the potential applicability of LLMs in alleviating this issue by demonstrating the assessment of various LLMs in recognizing bacterial host organisms and genetic toolbox, and evaluating the efficacy of annotated datasets for these entities derived from scientific literature in enhancing the models' predictive capabilities. Additionally, we introduce a web-based interface through which users can interact with the LLM and access answers augmented with external biological resources. We anticipate that users will utilize these tools to extract pertinent information from literature sources concerning biological entities and genetic tools and components, encompassing organism names and various tools such as promoters, plasmids, and phages, which are essential for the engineering of microorganisms.

Acknowledgments

This material is based upon work for the NLP for Synthetic Biology: Providing Generalizable Literature Knowledge Mining Through KBASE project supported by the U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research, under contract number DE-AC02-05CH11231 (LBL) and DE-SC-0012704 (BNL).

References

Kamil Albrycht, Adam A Rynkiewicz, Michal Harasymczuk, Jakub Barylski, and Andrzej Zielezinski. 2022. Daily reports on phage-host interactions. *Frontiers in Microbiology*, 13:946070.

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Merouane Debbah, Etienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, et al. 2023. Falcon-40b: an open large language model with state-of-the-art performance. *Findings of the Association for Computational Linguistics: ACL*, 2023:10755–10773.

A Arkin, N Baliga, J Braam, G Church, J Collins, R Cottingham, J Ecker, M Gerstein, P Gilna, J Greenberg, et al. 2010. Grand challenges for biological and environmental research: A long-term vision. Technical report.

Adam Arkin. 2008. Setting the standard in synthetic biology. *Nature biotechnology*, 26(7):771–774.

Adam P Arkin, Robert W Cottingham, Christopher S Henry, Nomi L Harris, Rick L Stevens, Sergei Maslov, Paramvir Dehal, Doreen Ware, Fernando Perez, Shane Canon, et al. 2018. Kbase: the united states department of energy systems biology knowledgebase. *Nature biotechnology*, 36(7):566–569.

Qijie Chen, Haotong Sun, Haoyang Liu, Yinghui Jiang, Ting Ran, Xurui Jin, Xianglu Xiao, Zhimin Lin, Hongming Chen, and Zhangmin Niu. 2023. An extensive benchmark study on biomedical text generation and mining with chatgpt. *Bioinformatics*, 39(9):btad557.

National Research Council, Division on Earth, Life Studies, Board on Life Sciences, Board on Chemical Sciences, Committee on Industrialization of Biology, and A Roadmap to Accelerate the Advanced Manufacturing of Chemicals. 2015. Industrialization of biology: A roadmap to accelerate the advanced manufacturing of chemicals.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2024. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36.

Drew Endy. 2005. Foundations for engineering biology. *Nature*, 438(7067):449–453.

Xinyu Gong, Jason Holmes, Yiwei Li, Zhengliang Liu, Qi Gan, Zihao Wu, Jianli Zhang, Yusong Zou, Yuxi Teng, Tian Jiang, et al. 2023. Evaluating the potential of leading large language models in reasoning biology questions. *arXiv preprint arXiv:2311.07582*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Dahyun Kim, Chanjun Park, Sanghoon Kim, Wonsung Lee, Wonho Song, Yunsu Kim, Hyeonwoo Kim, Yungi Kim, Hyeonju Lee, Jihoo Kim, et al. 2023. Solar 10.7 b: Scaling large language models with simple yet effective depth up-scaling. *arXiv preprint arXiv:2312.15166*.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. Biomistral: A collection of open-source pretrained large language models for medical domains. *arXiv preprint arXiv:2402.10373*.

Bridget T McInnes, J Stephen Downie, Yikai Hao, Jacob Jett, Kevin Keating, Gaurav Nakum, Sudhanshu Ranjan, Nicholas E Rodriguez, Jiawei Tang, Du Xiang, et al. 2022. Discovering content through text mining for a synthetic biology knowledge system. *ACS synthetic biology*, 11(6):2043–2054.

Meta-AI. 2024. Introducing meta llama 3: The most capable openly available llm to date. https://ai.meta.com/blog/llama3. Accessed: 2024-04-19.

Tomoko Mihara, Yosuke Nishimura, Yugo Shimizu, Hiroki Nishiyama, Genki Yoshikawa, Hideya Uehara, Pascal Hingamp, Susumu Goto, and Hiroyuki Ogata. 2016. Linking virus genomes with host taxonomy. *Viruses*, 8(3):66.

MosaicML-NLP-Team. 2023. Introducing mpt-30b: Raising the bar for open-source foundation models. Accessed: 2023-06-22.

Vivek K Mutalik, Joao C Guimaraes, Guillaume Cambray, Colin Lam, Marc Juul Christoffersen, Quynh-Anh Mai, Andrew B Tran, Morgan Paull, Jay D Keasling, Adam P Arkin, et al. 2013. Precise and reliable gene expression via standard transcription and translation initiation elements. *Nature methods*, 10(4):354–360.

Hiroki Nakayama, Takahiro Kubo, Junya Kamura, Yasufumi Taniguchi, and Xu Liang. 2018. doccano: Text annotation tool for human. Software available from https://github.com/doccano/doccano.

Matthew A Oberhardt, Raphy Zarecki, Sabine Gronow, Elke Lang, Hans-Peter Klenk, Uri Gophna, and Eytan Ruppin. 2015. Harnessing the landscape of microbial culture media to predict new organism–media pairings. *Nature communications*, 6(1):8493.

Morgan N Price and Adam P Arkin. 2017. Paperblast: text mining papers for information about homologs. *Msystems*, 2(4):10–1128.

Lorenz Christian Reimer, Joaquim Sardà Carbasse, Julia Koblitz, Christian Ebeling, Adam Podstawka, and Jörg Overmann. 2022. Bac dive in 2022: the knowledge base for standardized bacterial and archaeal data. *Nucleic Acids Research*, 50(D1):D741–D746.

Georges P Schmartz, Anna Hartung, Pascal Hirsch, Fabian Kern, Tobias Fehlmann, Rolf Müller, and Andreas Keller. 2022. Plsdb: advancing a comprehensive database of bacterial plasmids. *Nucleic Acids Research*, 50(D1):D273–D278.

Conrad L Schoch, Stacy Ciufo, Mikhail Domrachev, Carol L Hotton, Sivakumar Kannan, Rogneda Khovanskaya, Detlef Leipe, Richard Mcveigh, Kathleen

O'Neill, Barbara Robbertse, et al. 2020. Ncbi taxonomy: a comprehensive update on curation, resources and tools. *Database*, 2020:baaa062.

Gary Stacey. 2017. Grand challenges for biological and environmental research: Progress and future vision. a report from the biological and environmental research advisory committee. Technical report, USDOE Office of Science (SC), Washington, DC (United States). Biological and ....

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

WEF WEF. 2020. The global risks report 2020. In *Davos: World Economic Forum. Retrieved November*, volume 15, page 2020.

Brandon T Willard and Rémi Louf. 2023. Efficient guided generation for llms. *arXiv preprint arXiv:2307.09702*.

Huizi Yu, Lizhou Fan, Lingyao Li, Jiayan Zhou, Zihui Ma, Lu Xian, Wenyue Hua, Sijia He, Mingyu Jin, Yongfeng Zhang, et al. 2024. Large language models in biomedical and health informatics: A bibliometric review. *arXiv preprint arXiv:2403.16303*.

A A List of Genetic Toolbox Keywords

bacteria antibiotic resistance marker, bacteria auxotrophic vector, bacteria bicistronic design, bacteria biosensors, bacteria broad-host range, bacteria chassis, bacteria counter-selection, bacteria CRISPR toolbox, bacteria CRISPR tools, bacteria degradation tags, bacteria fluorescence reporter, bacteria fluorescent marker, bacteria Fosmid system, bacteria genetic elements, bacteria genetic engineering toolbox, bacteria genetic modification, bacteria genetic toolbox, bacteria golden gate parts, bacteria heterologous expression, bacteria inducible promoters, bacteria plasmid replicon, bacteria RBS part, bacteria recombineering tools, bacteria riboswitch, bacteria ribozyme, bacteria selection marker, bacteria shuttle vector, bacteria strain engineering, bacteria suicide vector, bacteria Tn5, bacteria Tn7, bacteria TnSeq, bacteria transformation method, bacteria transposons, bacterial conjugative, bacterial genetic parts, bacterial genetic tools, bacterial genome editing, bacterial inducer, bacterial integrative vector, bacterial molecular toolbox, bacterial plasmid collection, bacterial promoter library, bacterial replicative vector, bacterial reporter, bacterial transcription terminator, bacterial vectors, bglbrick, biobrick, counter-selection marker, Cre-lox, genetic toolkit, lambda red system, standard biological parts, synbio reporter system, synbio toolkit

REAL: A Retrieval-Augmented Entity Linking Approach for Biomedical Concept Recognition

Darya Shlyk[1*], **Tudor Groza**[2], **Stefano Montanelli**[1],
Emanuele Cavalleri[1] and **Marco Mesiti** [1]
[1]Università degli Studi di Milano, Via Giovanni Celoria 19, 20133 Milan, Italy
[2]School of Electrical Engineering, Computing and Mathematical Sciences,
Curtin University, Kent St, Bentley WA 6102, Australia

Abstract

Large Language Models (LLMs) offer an appealing alternative to training dedicated models for many Natural Language Processing (NLP) tasks. However, outdated knowledge and hallucination issues can be major obstacles in their application in knowledge-intensive biomedical scenarios. In this study, we consider the task of biomedical concept recognition (CR) from unstructured scientific literature and explore the use of Retrieval Augmented Generation (RAG) to improve accuracy and reliability of the LLM-based biomedical CR. Our approach, named REAL (Retrieval Augmented Entity Linking), combines the generative capabilities of LLMs with curated knowledge bases to automatically annotate natural language texts with concepts from bio-ontologies. By applying REAL to benchmark corpora on phenotype concept recognition, we show its effectiveness in improving LLM-based CR performance. This research highlights the potential of combining LLMs with external knowledge sources to advance biomedical text processing. Source code is available at: https://github.com/dash-ka/REAL-BioCR.

1 Introduction

Biomedical Concept Recognition (CR) aims to identify and link textual mentions of biomedical concepts to entries in expert-curated knowledge bases and ontologies. CR combines two subtasks from the standard information extraction pipeline: entity recognition (NER) and entity linking (EL), sometimes referred to as named entity disambiguation (NED) or grounding. NER aims to detect strings in text that refer to classes of biomedical entities, such as phenotypes, diseases, or genes. EL maps those strings to terms in an ontology, such as Human Phenotype Ontology (HPO) (Köhler et al., 2014) for phenotypic features, Mondo (Vasilevsky et al., 2020) for disease terms, and HGNC (Eyre et al., 2006) for human genes. Automated CR methods represent an active research area and are essential for a range of downstream biomedical applications. In genomic medicine, for instance, accurately recognizing phenotype concepts from free-text medical notes is the starting point to improve genetic disease diagnostics (Labbé et al., 2023).

State-of-the-art CR systems rely on fine-tuning transformer-based language models pretrained on biomedical texts, such as BioBERT (Lee et al., 2020), and have restricted scope, targeting a single or few application domains (Feng et al., 2022; Luo et al., 2021). The major limitation of these approaches is the need for domain-specific training with expert-labeled corpora, which is not always feasible due to the scarsity of annotated data in the biomedical field (Fries et al., 2022). On the other hand, general-purpose Large Language Models (LLM), such as OpenAI's Generative-Pretrained Transformer (GPT), have demonstrated remarkable zero and few-shot learning abilities, offering significant potential for biomedical NLP. Recent studies have shown promising results when using LLMs in clinical information extraction without domain-specific training (Agrawal et al., 2022; Meoni et al., 2023). However, challenges persist regarding factual accuracy in generated responses, hindering their usability for knowledge-intensive tasks in specialized domains (Gao et al., 2023; Reese et al., 2023). To address these challenges, Retrieval-Augmented-Generation (RAG) (Lewis et al., 2020) has been recently proposed as a technique to enhance LLMs with relevant information retrieved from external knowledge bases through semantic similarity calculation.

Our study aims to explore the application of the RAG paradigm in the context of biomedical CR. To this end, we developed REAL, a Retrieval-Augmented Entity Linking approach for ontology-based CR. To overcome the limitation of training dedicated NER and EL models, our approach lever-

[*]Corresponding author: darya.shlyk@unimi.it

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 380–389
August 16, 2024. ©2024 Association for Computational Linguistics

ages prompting techniques with general purpose LLMs to handle both tasks in a unified pipeline. Given a text, REAL first identifies mentions of concepts belonging to some target biomedical domain through a zero-shot NER, and then associates these mentions to terms in the domain ontology using retrieval-enhanced Entity Linking. By embedding the mention and ontology concepts into a common dense space, the retrieval mechanism provides the LLM with a selection of candidates from a bio-ontology identified through nearest neighbor search. By synergistically combining the retrieval mechanism with prompt-engineering, REAL aims to leverage up-to-date knowledge with existing knowledge bases, thereby improving the accuracy and reliability of LLM-based CR.

We summarize our contributions as follows:

- We propose a novel RAG-based approach that leverages general-purpose LLMs for automatic annotation of unstructured scientific literature with concepts from bio-ontologies. Our approach is versatile and can be easily adopted in various application domains without requiring domain-specific training.

- We conduct experiments with two benchmark corpora, studying the effectiveness of our approach on the phenotype concept recognition task. The results show that REAL can achieve competitive performance, indicating a great promise for the RAG paradigm in the context of biomedical concept recognition.

2 Related Work

2.1 Biomedical Concept Recognition

Biomedical CR tools predominantly rely on dictionary-based methods, using lexical matching with lookup tables. The OBO annotator (Taboada et al., 2014), the NCBO annotator (Jonquet et al., 2009), and the Monarch Initiative platform (Putman et al., 2023) are examples of tools that achieve high precision, but often suffer from low recall.

To overcome the limitations of dictionary-based methods, the recent research explored the use of neural-based models, with significant performance improvements. State-of-the-art approaches leverage pretrained BERT (Bidirectional Encoder Representations from Transformers) architectures. For instance, PhenoBert (Feng et al., 2022) implements a complex pipeline exploiting convolutional neural networks (CNNs) with BERT to automatically recognize HPO terms from free text. Phenotagger (Luo et al., 2021) is a hybrid approach that combines dictionary and deep learning methods. Specifically, Phenotagger fine-tunes a pretrained BioBERT model on weakly supervised datasets. These solutions necessitate task-specific training, requiring extensive computational resources and significant human effort for the manual annotation of large training corpora.

With the the advent of ChatGPT, researchers started to consider prompt-based approaches that leverage impressive language understanding capabilities of instruction-based generative models to address a wide spectrum of NLP tasks with no domain or task-specific training. One of the most prominent examples is SPIRES (Structured Prompt Interrogation and Recursive Extraction of Semantics) (Caufield et al., 2024) that leverages LLMs to assist the automatic construction of knowledge bases. Given an input text and a user-defined conceptual schema, the method recursively prompts an LLM to extract structured knowledge conforming with the schema's classes relevant for a given domain. The schema guides the LLM in extracting named entities that meet specific property constraints. To map extracted entities to ontology identifiers, SPIRES adopts the Ontology Access Kit library (OAKlib), which provides interfaces for external annotation tools, including the OBO annotator, and the Ontology Lookup Service.

2.2 Prompt-based Phenotyping

Several recent studies have employed prompt engineering techniques with LLMs to evaluate their capability in performing end-to-end phenotype concept recognition. Labbé et al. (2023) prompt GPT3.5 model to directly extract HPO term labels alongside corresponding IDs from medical texts. Their study highlights the limitations associated with purely prompt-based concept recognition, suggesting that potential improvements could be achieved by integrating factual knowledge from reference resources to aid in the generation process.

Groza et al. (2024) evaluated the OpenAI GPT-3.5 and GPT-4.0 models on phenotype concept recognition by testing alternative prompting strategies, including pipelined and in-context learning approaches. The former involves two sequential prompts: one for phenotype extraction and another for linking to HPO IDs. The latter approach incorporates the target subset of HPO label - ID pairs from the reference ontology inside the prompt as

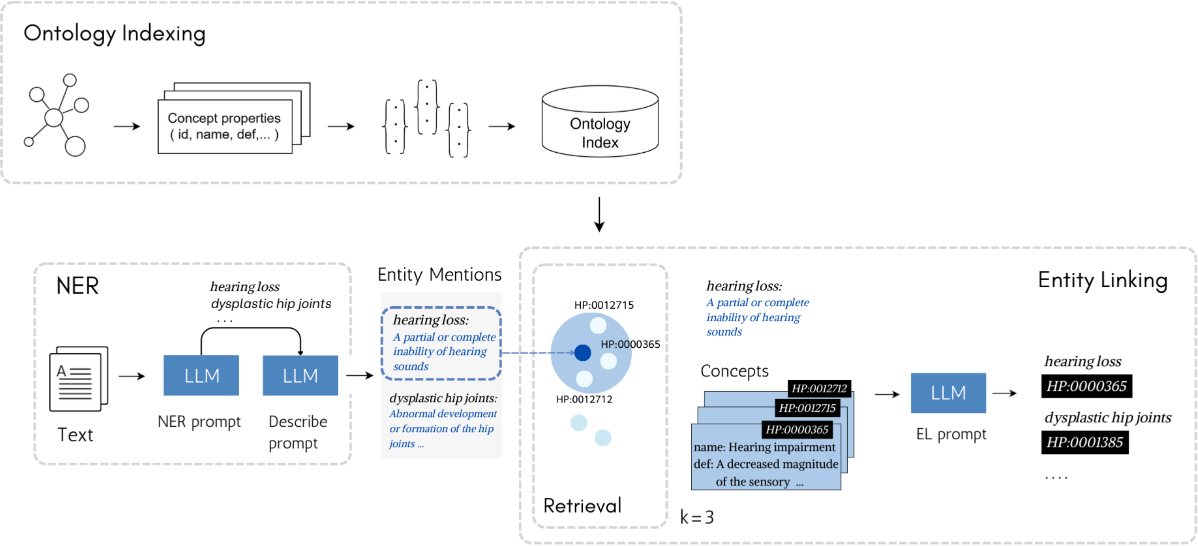


Figure 1: A high-level overview of the REAL approach.

context. Their findings demonstrate that in-context learning with pre-filtered ontology terms has the potential to surpass state-of-the-art CR systems.

The idea to couple parametric knowledge of an LLM with vast external knowledge repositories to improve the factuality and accuracy of the LLM responses forms the foundation of RAG. This technique involves chunking and embedding the knowledge resource into a set of vectors, followed by retrieving *top-k* relevant chunks based on semantic similarity with the user query, which are then incorporated into the LLM prompt. To the best of our knowledge, ours is the first work to explore the application of RAG in the field of biomedical concept recognition (CR). In REAL, we employ RAG to assist the LLM in linking textual mentions of biomedical entities to terms in domain ontologies.

3 Methodology

The ontology-based CR problem can be formally presented as two consecutive tasks, NER and EL, as follows. Let $\mathcal{O}$ denote a set of concepts $\{C_1, \ldots, C_n\}$ defined in the domain ontology used for text annotation. Given a text T, the NER task identifies textual mentions of biomedical entities from the target domain, $m_1, \ldots, m_h$. Then, the EL task consists in assigning each entity mention m_i to a concept $C \in \mathcal{O}$ that best represents it.

As shown in Figure 1, REAL implements CR in a pipeline consisting of three main phases: *Ontology Indexing*, *NER*, and *Retrieval-enhanced Entity Linking*. The ontology indexing is executed only once during the pre-processing to convert the concepts in $\mathcal{O}$ into a searchable index. The main workflow starts with the zero-shot NER (in the left bottom of the figure), where we prompt the LLM to extract instances of a specified entity type from T and generate a short definition for each of them. In the Retrieval-enhanced Entity Linking phase (right bottom part of the figure), we search in the ontology index the *top-k* most similar concepts to the embedding of m. They are the candidates for entity linking and the best matching is identified by properly instructing an LLM prompt. Details of the approach are provided in the remainder.

3.1 Ontology Indexing

To implement RAG for CR with the domain ontology $\mathcal{O}$ as a reference knowledge resource, we create vector embeddings for concepts in $\mathcal{O}$ and index them inside a vector store. This process creates an ontology index $\mathcal{I}$, that we can query to retrieve ontology concepts with the most similar embedding vector to the embedding of a given query.

In this study, we used ChromaDB[1], an open-source vector database, to store concept embeddings and perform semantic similarity search in the embedding space using the cosine similarity function. However, the proposed method can employ any database that enables efficient vector search capabilities. Unlike other vector stores, ChromaDB provides interfaces to popular LLM providers, and automatically computes the embedding from text using the specified embedding model. Specifically, this study uses the OpenAI

[1] https://www.trychroma.com/

Concept prompt

```
As a clinical expert, write a single sentence
definition that explains the meaning of the
concept: { concept name }
```

Figure 2: Prompt template for generating a definition for a given concept name.

`text-embedding-ada-002` model for concept embedding. We store the computed embeddings alongside the concept properties inside the index $\mathcal{I}$. To construct input texts for the embedding model, we employ the concept name and its definition provided among concept properties in a given ontology. This design choice stems from the need to create a vector representation that captures the meaning of a concept, with the concept name and definition providing the minimal necessary information to achieve this goal. Whenever a textual definition is not available in the concept properties, we automatically generate one from the concept name by prompting an LLM using the prompt in Figure 2 (the variable part of the prompt is colored in green).

Example 1 *Suppose we are interested in creating an ontology index for the HPO. A vectorial representation for each HPO concept is generated by concatenating its name and definition and stored into ChromaDB with other concept properties. For instance, for the HPO term "Breast carcinoma" (id:* `HP:0003002`*), the following text has been used for generating the concept embedding:*

```
name: Breast carcinoma
definition: Presence of carcinoma in breast
```

The prompt template in Figure 2 is used to generate a single sentence definition for 2,586 HPO terms that do not have a definition in the ontology.

3.2 NER

Given the input text T and the specification of the target biomedical entities to be extracted, the NER step produces a set of pairs $\mathcal{P} = \{(m_i, d_i) \mid 0 \leq i \leq h\}$, where m_i represents a mention of the target entity extracted from T, and d_i denotes a concise definition for that entity mention. This step employs zero-shot prompting with LLMs using two consecutive prompts: the *NER prompt* for entity extraction and the *Describe prompt* for definition generation. The NER prompt in Figure 3 incorporates the input text T and directs the LLM to extract spans in T that represent instances of the target entity type defined for a given application domain. The domain adaptation of the NER task is performed by changing the type of the target biomedical entity specified in the prompt, e.g., genes, phenotypes. The Describe prompt in Figure 4 operates on the list of entity mentions $m_1, \ldots, m_h$, produced with the NER prompt, and tasks the LLM to generate a definition for each extracted mention. Besides the list of entity mentions, this prompt also includes the original text T to help the LLM to compose contextually informed entity definitions.

NER prompt

```
From the text below, extract all mentions of
the following entities in the following format:
entities : < a semicolon-separated list of noun
phrases containing a mention of { target entities }
It must be semicolon-separated.
Split entities containing words "and", "," into
separate entities.>

Text:
{ input text }
```

Figure 3: Prompt template for NER.

Describe prompt

```
Given a text and a semicolon-separated list of
entities from that text, write a definition for
each entity in the following format:
<entity>: < a detailed definition that explains
the meaning of the entity in one sentence >

Here is the text:
{ input text }

Here are the entities:
{ entity mentions }
```

Figure 4: Prompt template for entity description.

Example 2 *Let T be the following text:*

```
The combination of either the skin tumours
or multiple odontogenic keratocysts.
```

Suppose we are interested in extracting mentions of human phenotypes from T. Then, in the NER prompt, we specify the following target entities: "human phenotypes, including physical abnormalities, symptoms of disease, and inherited disorders". Given T, the NER prompt extracts the entities: `skin tumors` *and* `odontogenic keratocysts`.

For each of them (and considering T) a definition is generated using the prompt in Figure 4 as follow:

skin tumors: Abnormal growths or masses that occur in the skin and can be benign or malignant.

odontogenic keratocysts: Cysts that develop in the jawbones and are derived from the remnants of dental tissue.

3.3 Retrieval Augmented Entity Linking

Given the Ontology index $\mathcal{I}$, and a set $\mathcal{P}$ of pairs (m, d) obtained as a result of the NER phase, for each element in $\mathcal{P}$, the EL phase is realized in two steps: *Candidate Retrieval* and *Entity Linking*.

3.3.1 Candidate Retrieval

Given the entity mention m with its definition d and a user-defined parameter k, we first embed (m, d) into the same embedding space with ontology concepts, and then retrieve *top-k* semantically similar concepts $\{C'_1, \dots C'_k\}$ from $\mathcal{I}$ by approximate k-nearest neighbor search. We adopt the same embedding strategy for ontology concepts to enable consistent representation of the entity mentions in the common vector space (see Section 3.1).

Example 3 *Consider the first entity mention identified in Example 2. The following text is used to compute the mention embedding q:*

```
name: skin tumors
definition: Abnormal growths or masses
that occur in the skin and can be benign
or malignant.
```

By querying $\mathcal{I}$ with q, we can retrieve the top-3 concepts in HPO with the highest similarity score, according to cosine similarity function:

- ID: HP:0008069
  name: Neoplasm of the skin
  definition: A tumor (abnormal growth of tissue) of the skin.
  score: 0.9329

- ID: HP:0000951
  name: Abnormality of the skin
  definition: An abnormality of the skin.
  score: 0.8974

- ID: HP:0012056,
  name: Cutaneous melanoma
  definition:The presence of a melanoma of skin.
  score: 0.8937

```
EL prompt

As an expert clinician, your task is to accurately
identify the { domain ontology } concept mentioned
in the provided text using the concepts listed below.
Accuracy is paramount. If the text does not precisely
refer to any of the concepts listed below, please
return "None"; otherwise, return the corresponing
concept ID in the following format:
answer: ‹ concept ID or None ›
confidence: ‹ one of: HIGH, LOW, MEDIUM ›

Here are some examples:
{ examples }

Below are the concepts:
{ candidate concepts }

 Text:
{ entity description }
```

Figure 5: Prompt template for EL.

3.3.2 Entity Linking

To ground m using the retrieved candidate set $\{C'_1, \dots C'_k\} \subset \mathcal{O}$, we re-frame the EL task as a multiple-choice selection and prompt the LLM to identify the ontology concept among the provided candidates that best matches an entity description (m, d). The candidate concepts are provided as part of the prompt with their properties, including concept ID, name and definition. When selecting a concept for a given mention m, the LLM is instructed to associate a confidence level with its answer (a value in $\{\mathtt{HIGH}, \mathtt{MEDIUM}, \mathtt{LOW}\}$), which we use as a filtering mechanism when parsing the EL results. The EL prompt is generated according to the template in Figure 5, and adopts a few-shot learning technique, where the LLM learns to perform the EL task in-context by following a set of examples provided as part of the prompt.

Examples are ontology-specific and present the following structure: i) a list of concepts with the ID, name, and definition; ii) a text describing the entity mention to be grounded; iii) the expected answer; and iv) the associated confidence level. In the case of the HPO, Figure 6 shows the example that can be included in the EL prompt for one-shot Entity Linking. This negative example serves the purpose of instructing the LLM to be conservative and refrain from mapping any entity mention extracted with NER unless the matching concept belongs to the provided candidates.

```
[Concept A]
ID: HP:0034057
name: Fetal anomaly
definition: Structural or functional abnormalities of the fetus.

[Concept B]
ID: HP:0034058
name: Abnormal fetal morphology
definition: Any structural anomaly of the fetus.

[Concept C]
ID: HP:0034059
name: Abnormal fetal physiology
definition: Any functional anomaly of the fetus.

Text:
name: physical abnormalities
definition: Structural or functional abnormalities in
the body that can be observed or measured.


answer: None
confidence: HIGH
```

Figure 6: An HPO-specific example for the EL task.

Example 4 *Consider the mention and the set of retrieved candidates in Example 3. The template in Figure 5 is filled with: the name of the domain ontology (HPO); the HPO-specific example in Figure 6; the retrieved candidates and their properties; the description of the entity to be grounded. Invoking the LLM with EL prompt, the following result is returned:*

```
answer: HP:0008069
confidence: HIGH
```

4 Experiments

4.1 Benchmark Corpora

To validate our approach, we evaluate the performance of REAL for clinical phenotyping and phenotype annotation using two publicly available benchmark datasets: the HPO GSC+ (Lobo et al., 2017) and the dev component of the corpus published by BioCreative VIII Track 3 (Weissenbacher et al., 2023), referred to as BIOC-GS hereafter. HPO GSC+ consists of 228 manually annotated PubMed abstracts, with a total of 1933 annotations that cover 497 unique HPO IDs. The BIOC GS consists of 454 clinical observations manually annotated for phenotypes identified during dysmorphology physical examinations, that cover a total of 358 unique HPO IDs. As a reference resource for grounding, we use HPO, that provides a standardized vocabulary of phenotypic abnormalities associated with human hereditary and other diseases (Köhler et al., 2019). After preprocessing the ontology file, we indexed a total of 18.536 HPO concepts (See Section 3.1).

4.2 Experimental Setting

Currently, the REAL implementation relies on the OpenAI GPT models and feeds the prompts to the LLM by calling the OpenAI API. For evaluation, we use the `gpt-3.5-turbo-16k` model accessed through the GPT-3 completion endpoint, with default settings for temperature and `max tokens`. The number of LLM calls per document is estimated as follows: 2 requests sent to the OpenAI completion API endpoint[2] in the NER step, one for entity extraction and one for definition generation. Followed by h calls in the EL step, one call for each extracted mention. Additionally, for candidate retrieval, each entity mention requires a call to the OpenAI embedding API endpoint[3], which is handled automatically by the ChromaDB vector store. Due to constraints on the context window size (16,385 tokens for `gpt-3.5-turbo-16k` model), we limit the retrieved candidate set to a small number. In our experiments, we set $k = 3$, and included three candidate concepts in the EL prompt, as we observed no substantial improvements when using a larger number of candidates (see Section 4.4 for further discussion). Moreover, to ensure precise results in the EL phase, we opt to consider only mention/concept pairs associated with a `HIGH` confidence level, discarding less confident answers generated by the LLM.

To evaluate the effectiveness of the RAG paradigm in the context of the LLM-based biomedical concept recognition, we benchmark against a base case, where we directly instruct the GPT-3.5 model to extract and align HPO concepts from the input text using a single instructional prompt. The baseline prompt used in the experiments is adopted from Groza et al. and reported in Appendix 8.

In assessing the role of the LLM component in the entity linking step, our evaluation involves two distinct grounding strategies: one relies on the LLM to select the appropriate candidate concept for a given entity mention, while the other always selects the first matching concept retrieved by the embedding-based search. We refer to this latter strategy, which does not utilize the LLM in the linking phase, as *REAL-1st HIT* to differentiate it from the strategy using GPT3.5 for grounding, which we denote as *REAL-GPT3.5*. For a fair comparison with existing unsupervised methods for concept recognition, our evaluation in-

[2] `https://api.openai.com/v1/chat/completions`

[3] `https://api.openai.com/v1/embeddings`

	Document level			Mention level		
System	P	R	F1	P	R	F1
GPT-3.5	0.12	0.28	0.16	0.07	0.17	0.10
SPIRES	**0.84**	0.31	0.45	**0.84**	0.19	0.31
REAL-1st hit	0.40	**0.49**	0.44	0.33	**0.36**	0.39
REAL-GPT3.5	0.68	0.48	**0.56**	0.67	0.32	**0.43**

Table 1: Evaluation results on HPO GSC+

	Document level			Mention level		
System	P	R	F1	P	R	F1
GPT-3.5	0.26	0.33	0.29	0.22	0.29	0.25
SPIRES	**0.93**	0.31	0.47	**0.93**	0.19	0.47
REAL-1st hit	0.59	0.49	0.42	0.59	0.48	0.41
REAL-GPT3.5	0.69	**0.67**	**0.66**	0.68	**0.66**	**0.65**

Table 2: Evaluation results on BIOC GS

cludes SPIRES, a close prompt-based alternative to the REAL approach, accessible with local installation of the OntoGPT Python package [4]. For entity linking, SPIRES uses the OBO annotator (Taboada et al., 2014), a state-of-the-art dictionary-based method, designed for automatic annotation of biomedical literature with HPO terms. To execute HPO concept recognition with SPIRES, we utilize a predefined template for extracting human phenotypes, which is provided with the OntoGPT installation.[5] For phenotype extraction, SPIRES uses the `gpt-3.5-turbo-16k` model.

We evaluate the results by computing standard metrics for the concept recognition task: precision (P), recall (R) and F1-score (F1). The evaluation is performed at both document and mention levels. At the document level, we compute true positives as a set of target concepts that were found at least once in a given document and assigned a correct HPO identifier. At the mention-level, we account for all occurrences of a target concept within a document.

4.3 Results

Tables 1 and 2 present the evaluation results for the HPO concept recognition on HPO GSC+ and BIOC GS datasets, respectively. To facilitate the performance comparison across systems, Figure 7 illustrates the precision, recall, and F1 score values considering the document level evaluation, which closely reflects the pattern observable at the mention level. Consistent results are also observed when conducting testing on the two datasets, as discussed in this section. Among other methods, *REAL-GPT3.5* can correctly recognize more HPO concepts, achieving the best F1 scores at both mention and document level. The dictionary matching method used with the OBO annotator, allows SPIRES to achieve the highest precision, which does not compensate for poor recall rates. The results show that the retrieval mechanism integrated in REAL significantly improves the recall, compared to other methods. In fact, *REAL-1st hit*, that uses the 1st retrieved concept for entity linking, achieves similar F1 score as SPIRES while balancing the precision and recall rates. Comparing the two grounding strategies, we observe that *REAL-GPT3.5* improves the precision over *REAL-1st hit* at both mention and document level. Leveraging the LLM for entity linking produces more precise results as it enables reasoning over the best match through multiple-choice selection and effectively filters out spurious extractions, that is, entity mentions erroneously identified as phenotypes by NER. In summary, the results on the GSC+ and BIOC GS datasets demonstrate the effectiveness of the REAL approach for phenotype concept recognition. Our experiments here were limited to GPT-3.5, but it is likely that GPT-4 will yield even better results.

4.4 Error analysis and discussions

The formulation of the NER prompt represents one of the critical aspects for the success of the approach. Poor results on NER propagate down the pipeline affecting the usefulness of the entity linking step. We assess the completeness of the NER results through a manual analysis of the generated extractions. Our evaluation suggests that the NER prompt achieves pertinent extractions providing a comprehensive coverage of the phenotypic features in both corpora. Some extractions from the HPO GSC+ include concepts not covered in the HPO ontology, such as mentions of diseases (*Prader-Willi syndrome, Angelman syndrome*), or generic phenotype-related concepts (*human anomaly*, *genetic abnormalities*). Additionally, we observe that a number of HPO terms extracted by REAL lack annotation in HPO GSC+. For instance, the phenotype *"Uniparental disomy"* is recognized 17 times in the corpus, but it is not present in the gold standard annotations, despite the existence of the exact match in the HPO: *"Uniparental disomy"* (`HP:0032382`). Such extractions represent the main cause of false positives and frequently in-

[4] `https://github.com/monarch-initiative`

[5] `https://github.com/monarch-initiative/ontogpt/blob/main/src/ontogpt/templates/`

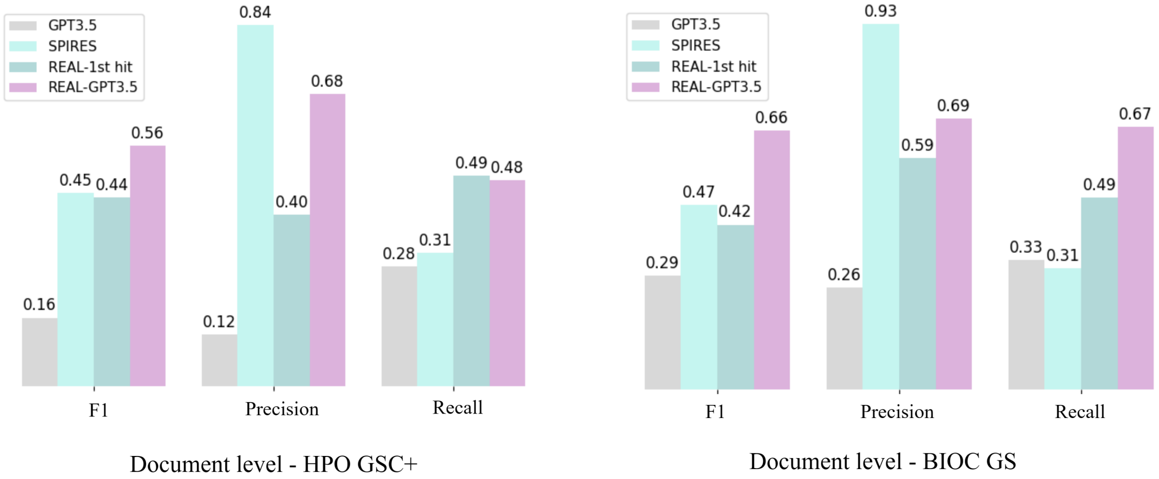


Figure 7: Document level evaluation results on HPO GSC+ and BIOC GS.

volve HPO terms close to the root of the taxonomic tree, such as *phenotypic abnormality (*`HP:0000118`*)* and *mode of inheritance* (`HP:0000005`).

Upon examining the frequently missed HPO terms, we identified two main causes of false negatives. The first issue is specific to HPO GSC+ and stems from the overlapping concepts, where phrases contain multiple nested HPO terms. For instance, the phrase *"skin tumors"* is annotated with both *"Neoplasm of the skin"* (`HP:0008069`) and *"Neoplasm"* (`HP:0002664`). By design, REAL, extracts mentions of entities as a whole and annotates to the most specific HPO term, failing to produce identifiers for nested concepts. This explains a high number of omissions for generic terms such as, *"Nurofibroma"*, *"Schwannoma"*, and *"Meningoma"*, usually nested within more specific HPO concepts.

The second issue involves complex entity mentions, frequently found in clinical notes, that have a form of compound and prepositional phrases, such as, *"scarring between 2 and 3"* and *"2,3 syndactyly bilaterally in feet"*. These extractions may produce definitions where the meaning of the entity is altered with respect to the target HPO term, yielding a poor set of retrieved candidates. For instance, the extracted mention, *"scars on axillary lines bilaterally"* produces a definition (*"Permanent marks or blemishes that have formed on the skin in the areas of the armpits, appearing on both sides of the body."*), that shifts the entity's meaning away from the target concept , *"Scarring"* (`HP:0100699`), towards related HPO terms, such as *"Axillary freckling"*, and *"Axillary lymphadenopathy"*. Moreover, due to the high level of granularity of the extracted mentions in the BIOC GS dataset, entities are often grounded in HPO terms that are more specific than those provided in the annotations. For example, the mention *"skins on the right foot feet thickend"* is mapped to *"Hypertrophy of skin of soles"*(`HP:0007403`) (i.e., *"Thick skin of soles"*), instead of the target *"Thickend skin"* (`HP:0001072`). These and similar issues can arise as a consequence of annotation idiosyncrasies that vary across benchmarks and could be addressed via additional post-processing of the NER results.

By analyzing the retrieval results, we found that around 65% of target concepts in HPO GCS+ (78% in BIOC GS) were effectively retrieved among candidates using approximate k-nearest neighbor search with $k = 3$. Preliminary experiments with greater values of k show no significant improvement, suggesting that the effectiveness of the candidate retrieval step mostly depends on the ability of the LLM to produce entity descriptions that are semantically close to target HPO terms. Our approach relies on the LLM to produce factual definitions for extracted mentions. However, future research might explore alternative strategies to ensure the factuality of the generated definitions. (Remy and Demeester, 2023).

It is important to stress that the domain expertise requirements vary across different phases of the concept recognition pipeline. In the grounding step using RAG, the domain knowledge is provided from outside, significantly reducing the expertise required by the LLM for entity normalization. In contrast, the biomedical NER task relies on the domain knowledge encoded within the model's parameters, demanding greater familiarity with the target domain to accurately recognize and define entities. This makes the NER task more knowledge-intensive and crucial for the overall success of the approach.

5 Concluding remarks

In this work, we introduced a novel approach for ontology-based concept recognition, that leverages RAG to harness general-purpose LLMs for automatic annotation of biomedical texts with classes from domain ontologies. The approach does not require domain specific training, but relies on prompt-engineering for both NER and EL tasks, integrating a retrieval mechanism to dynamically source domain knowledge from biomedical ontologies. We discussed the effectiveness of our approach on clinical phenotyping and phenotype annotation with experiments conducted on HPO GSC+ and BIOC GS benchmark corpora. Ongoing efforts focus on refining the prompt design to enhance performance and consider the integration with other GenAI providers. Using GPT models through OpenAI's API hinders the reproducibility of the results, which represents the main limitation of the current implementation. We plan to address this issue using a local installation of open-source LLMs. Furthermore, future research activities include conducting a comprehensive cross-domain evaluation to assess the generalizability of the proposed solution to diverse application domains.

Acknowledgments

This research was supported by: the "National Center for Gene Therapy and Drugs based on RNA Technology", PNRR-NextGeneration EU program [G43C22001320007]; the MUSA - Multilayered Urban Sustainability Action - Project, funded by the PNRR-NextGeneration EU program ([G43C22001370007], Code ECS00000037).

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. *arXiv preprint arXiv:2205.12689*.

J Harry Caufield et al. 2024. Structured Prompt Interrogation and Recursive Extraction of Semantics (SPIRES): a method for populating knowledge bases using zero-shot learning. *Bioinformatics*, 40(3):btae104.

Tina A Eyre, Fabrice Ducluzeau, Tam P Sneddon, Sue Povey, Elspeth A Bruford, and Michael J Lush. 2006. The HUGO gene nomenclature database, 2006 updates. *Nucleic acids research*, 34(suppl_1):D319–D321.

Yuhao Feng, Lei Qi, and Weidong Tian. 2022. PhenoBERT: a combined deep learning method for automated recognition of human phenotype ontology. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 20(2):1269–1277.

Jason Alan Fries, Natasha Seelam, Gabriel Altay, Leon Weber, Myungsun Kang, Debajyoti Datta, Ruisi Su, Samuele Garda, Bo Wang, Simon Ott, et al. 2022. Dataset debt in biomedical language modeling. In *Challenges & Perspectives in Creating Large Language Models*.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Tudor Groza, Harry Caufield, Dylan Gration, Gareth Baynam, Melissa A Haendel, Peter N Robinson, Christopher J Mungall, and Justin T Reese. 2024. An evaluation of GPT models for phenotype concept recognition. *BMC Medical Informatics and Decision Making*, 24(1):30.

Clement Jonquet, Nigam H Shah, Cherie H Youn, Mark A Musen, Chris Callendar, and Margaret-Anne Storey. 2009. NCBO annotator: semantic annotation of biomedical data. In *Int'l Semantic Web Conf., Poster and Demo Session (ISWC 2009)*, 171.

Sebastian Köhler, Leigh Carmody, Nicole Vasilevsky, Julius O B Jacobsen, Daniel Danis, Jean-Philippe Gourdine, Michael Gargano, Nomi L Harris, Nicolas Matentzoglu, Julie A McMurry, et al. 2019. Expansion of the human phenotype ontology (HPO) knowledge base and resources. *Nucleic acids research*, 47(D1):D1018–D1027.

Sebastian Köhler, Sandra C Doelken, Christopher J Mungall, Sebastian Bauer, Helen V Firth, Isabelle Bailleul-Forestier, Graeme CM Black, Danielle L Brown, Michael Brudno, Jennifer Campbell, et al. 2014. The human phenotype ontology project: linking molecular biology and disease through phenotype data. *Nucleic acids research*, 42(D1):D966–D974.

Thomas Labbé, Pierre Castel, Jean-Michel Sanner, and Majd Saleh. 2023. ChatGPT for phenotypes extraction: one model to rule them all? In *45th Annual Int'l Conf. of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 1–4. IEEE.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Manuel Lobo, Andre Lamurias, Francisco M Couto, et al. 2017. Identifying human phenotype terms by combining machine learning and validation rules. *BioMed Research International*, 2017.

Ling Luo, Shankai Yan, Po-Ting Lai, Daniel Veltri, Andrew Oler, Sandhya Xirasagar, Rajarshi Ghosh, Morgan Similuk, Peter N Robinson, and Zhiyong Lu. 2021. PhenoTagger: a hybrid method for phenotype concept recognition using human phenotype ontology. *Bioinformatics*, 37(13):1884–1890.

Simon Meoni, Eric De la Clergerie, and Theo Ryffel. 2023. Large language models as instructors: A study on multilingual clinical entity extraction. In *Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 178–190.

Tim E Putman, Kevin Schaper, Nicolas Matentzoglu, Vincent P Rubinetti, Faisal S Alquaddoomi, Corey Cox, J Harry Caufield, et al. 2023. The Monarch Initiative in 2024: an analytic platform integrating phenotypes, genes and diseases across species. *Nucleic Acids Research*, 52(D1):D938–D949.

Justin T Reese, Daniel Danis, J Harry Caulfied, Elena Casiraghi, Giorgio Valentini, Christopher J Mungall, and Peter N Robinson. 2023. On the limitations of large language models in clinical diagnosis. *medRxiv*.

François Remy and Thomas Demeester. 2023. Automatic glossary of clinical terminology: a large-scale dictionary of biomedical definitions generated from ontological knowledge. *arXiv preprint arXiv:2306.00665*.

Maria Taboada, Hadriana Rodríguez, Diego Martínez, María Pardo, and María Jesús Sobrido. 2014. Automated semantic annotation of rare disease cases: a case study. *Database*, 2014:bau045.

Nicole Vasilevsky, Shahim Essaid, Nico Matentzoglu, Nomi L Harris, Melissa Haendel, Peter Robinson, and Christopher J Mungall. 2020. Mondo disease ontology: harmonizing disease concepts across the world. In *CEUR Workshop Proceedings, CEUR-WS*, volume 2807.

Davy Weissenbacher, Siddharth Rawal, Xinwei Zhao, Jessica RC Priestley, et al. 2023. PhenoID, a language model normalizer of physical examinations from genetics clinical notes. *medRxiv*.

A Appendix

Baseline prompt

~~~
You will be provided with a text in triple backticks.
The task is to perform automated concept recognition
using the Human Phenotype Ontology and extract all
Human Phenotype Ontology concepts found in the text.
Include the HPO ID of the concepts you find in the
result.

``` { input text } ```
~~~

Figure 8: Baseline prompt for HPO CR.

Is That the Right Dose? Investigating Generative Language Model Performance on Veterinary Prescription Text Analysis

Brian Hur[1] **Lucy Lu Wang**[1,2] **Laura Hardefeldt**[3] **Meliha Yetsigen**[1]
[1]University of Washington [2]Allen Institute for AI [3]University of Melbourne
{hurb, lucylw, melihay}@uw.edu, laura.hardefeldt@unimelb.edu.au

Abstract

Optimizing antibiotic dosing recommendations is a vital aspect of antimicrobial stewardship (AMS) programs aimed at combating antimicrobial resistance (AMR), a significant public health concern, where inappropriate dosing contributes to the selection of AMR pathogens. A key challenge is the extraction of dosing information, which is embedded in free-text clinical records and necessitates numerical transformations. This paper assesses the utility of Large Language Models (LLMs) in extracting essential prescription attributes such as dose, duration, active ingredient, and indication. We evaluate methods to optimize LLMs on this task against a baseline BERT-based ensemble model. Our findings reveal that LLMs can achieve exceptional accuracy by combining probabilistic predictions with deterministic calculations, enforced through functional prompting, to ensure data types and execute necessary arithmetic. This research demonstrates new prospects for automating aspects of AMS when no training data is available.

Clinical Note Inputs

Consultation Note:
History: Realised on Saturday that there was a wound on left fore leg...
Examination: Not limping in consult room...1cm wound that has been filled with granulation tissue. Swelling approx 0.5cm
Assessment: Wound over left fore limb...
Plan: Recheck in a week.

Prescription Label Information:
Item Label: Dog 14.00 x Amoxyclav Tabs 250Mg Give half tablet twice a day
Item Name: Amoxyclav Tabs 250Mg (100)
Weight: 10kgs
Units Dispensed: 14.0

Inferred Labels for Evaluation

Ingredient: Amoxicillin Clavulanate
Indication: Traumatic Injury
Frequency: 2 (daily doses)
Medication Size: 250 (mgs)
Dose Unit Size: 0.5 (tablets per dose)
Duration: 14 (days)
Dose: 12.5 (mg/kg)

Figure 1: Example of consultation and prescription note along with inferred labels.

1 Introduction

Antimicrobial resistance (AMR) has become a major public health concern, as antimicrobials are steadily losing their effectiveness in combating bacterial infections (O'Neill, 2016). AMR is not limited to human medicine; it is also a growing issue among animals (Ekakoro et al., 2022; Cummings et al., 2015), who can acquire and transmit multidrug-resistant pathogens to humans (Guardabassi et al., 2004). Antimicrobial Stewardship (AMS), which has demonstrated effectiveness in improving antimicrobial use in both human and animal healthcare (Davey et al., 2017; Hardefeldt et al., 2022), aims to optimize antimicrobial use to curtail the development and spread of AMR. Accurate dosing is part of this strategy, as overdosing can lead to toxicity and under-dosing can be particularly perilous as it can select for AMR organisms and lead to poor therapeutic outcomes (Roe et al., 2012; Grill and Maganti, 2011). A pragmatic way to improve dosing accuracy and optimizing antimicrobial use is through decision support systems in clinical settings (Hardefeldt et al., 2018b,a), where targeted dosing recommendations can be made in real-time.

Recent developments in Large Language Models (LLMs) introduce compelling opportunities for automated information extraction and decision support (Bubeck et al., 2023; Nori et al., 2023), as these models obviate the need for extensive labeled data (Brown et al., 2020). Such models can potentially furnish clinicians with data-driven counsel on optimal antimicrobial selection, treatment du-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 390–397
August 16, 2024. ©2024 Association for Computational Linguistics

ration, and dosing intervals, tasks that have been historically reliant on extensive labor-intensive labeled data compilation (Uzuner et al., 2010; Tao et al., 2017). To realize the potential for LLMs for extracting prescription elements, an essential step is empirical assessment of their ability to accurately extract relevant information from clinical text. Given the idiosyncratic nature of LLM training, which leverages instruction tuning rather than conventional training paradigms, it becomes vital to also scrutinize configuration variances for performance optimization (Zheng et al., 2023). Additionally, while the task of extracting elements out of prescriptions was explored in shared tasks such as the 2010 i2b2 challenge (Uzuner et al., 2011), these studies only evaluate the ability to extract text spans without performing numerical conversion. Converting text spans to numerical representations and performing necessary calculations to understand the dose and duration of a given medication are also essential to optimize antimicrobial use. Studies performing such numerical conversions rely on rules-based methods which are notoriously brittle, and only one study we identified made the corresponding algorithm available (Karystianis et al., 2016).

We leverage the VetCompass Australia (McGreevy et al., 2017) corpus, which comprises over 50 million clinical notes from over 200 veterinary clinics across Australia, as our primary data source. Our goal is to extract key information such as the active ingredient, the indication for antimicrobial use, and the dose and duration of the therapy. We assess the performance of LLMs in zero-shot and few-shot learning scenarios for extracting this critical information. By exploring the feasibility of applying LLMs to the VetCompass dataset, we seek to understand their potential in aiding dosing recommendations to support AMS. Specifically:

- We construct a veterinarian-labeled evaluation dataset of 200 clinical notes to study medication dosage extraction from veterinary notes;
- Using silver labels generated by a baseline BERT-based ensemble model to provide training examples, we benchmark the performance of LLMs against the baseline model for extracting medication dosage information, their indications, and active ingredients;
- While we demonstrate LLMs' proficiency in element extraction for dose and duration calculations, they falter at arithmetic operations crucial for deriving these elements (Yuan et al., 2023). We introduce methods to overcome this using functional prompting to combine the probabilistic predictions from the LLMs with deterministic calculations for labelling dosing elements in zero- or few-shot settings.[1]

2 Task & Dataset

We investigate the task of dose information extraction from veterinary clinical notes. Given textual clinical notes, the task is to extract seven labels, including five entity labels: active ingredient, clinical indication, frequency, medication size, and dosage unit size, along with two derived labels: dose and duration. A sample clinical note, prescription label, and the inferred target labels are illustrated in Figure 1. Prescription label information is provided as an input for all extractions except indication, for reasons of document length. For indication, the model is also given the set of potential indications; for ingredient, the set of potential ingredients. We evaluate accuracy based on exact match between the output label and the ground truth label.

Data Extraction and Label Creation We assemble a subset of 1500 clinical records sourced from VetCompass Australia (McGreevy et al., 2017), focusing on cases where patients received oral antimicrobial treatments as outlined in Hur et al. (2019). To facilitate further calculations, the patient's weight in kilograms and the total quantity of medication dispensed are also extracted from structured textual fields within the clinical records (Appendix Table 3). We extract the inferred label elements shown in Figure 1 using RxVetBERT, an ensemble model introduced in prior work (Hur et al., 2020, 2022); these inferred labels are used as silver labels for in-context learning examples. The extracted records and labels are partitioned into 1000 records for training, 300 for development, and 200 for test. The test set is reserved exclusively for the final evaluation stage after all prompts have been refined and optimized.

Gold Test Set Annotations To ensure label accuracy in the test set, two expert veterinarians manually annotated the data. Inter-annotator agreement was evaluated using exact match F1 scores. Initial IAA F1 was 0.8 for Indication, 0.985 for Dose, and

[1] The code and select models used in this study available at https://github.com/havocy28/prescription-text-analyzer

1.0 for Duration, Ingredient, Medication Unit Size and Dose Unit Size. High agreement in dosage and ingredient categories was due to their objectivity. Consensus on indication is more challenging, particularly when multiple clinical events complicated interpretation—e.g., in scenarios involving post-operative complications following traumatic injury, the indication could be correctly interpreted as either the initial injury or subsequent complications. Any annotation discrepancies were resolved through consensus discussion.

Indication and Ingredient Labels Indication labels are based on a subset of Veterinary Nomenclature (VeNOM) codes, a specialized adaptation of SNOMED for veterinary medicine (Brodbelt, 2019). We use the subset of 52 curated by (O'Neill et al., 2019), of which 23 appear in our test set. Ingredient labels are based on unique antimicrobial agents from VetCompass, which consist of 49 unique ingredients, 9 of which occur in our test set.

Dosing Elements Frequency indicates the amount of times per day a dose is given. It must be a numerical value such that it can be used in dosing calculations (e.g., 'twice daily' is converted to 2). Dose unit size indicates the amount of medication and must also be converted into a numerical value (e.g., 'half of a tablet' is converted to 0.5).

The medication dose and duration can then be calculated using the formulae:

$$\text{Dose} = \frac{D \times M}{W} \qquad \text{Duration} = \frac{T}{F \times D}$$

where D is the Dose Unit Size (number of tablets or volume of liquid), M the Medication Size (tablet size [in mg]), W the Weight of Patient (in kg), T the Total Units Dispensed, and F the Administration Frequency. The dose calculation is designed to tailor the medication dose to the individual's mass to achieve the optimal therapeutic efficacy while minimizing the risk of toxicity. This is particularly important for veterinary and pediatric patients, where the difference in mass between patients can vary greatly (Waldman et al., 2008).

3 Methodology

We benchmark three LLMs on this task: `GPT-3.5` (Brown et al., 2020), `GPT-4` (OpenAI, 2023a), and `LLAMA2-70B` (Touvron et al., 2023), against the baseline ensemble model (`RxVetBERT`), which combines rule-based methods and `VetBERT` as described in previous works (Hur et al., 2020, 2022).

Prompt Settings We compare the following:

Zero-shot: Utilizes text from the clinical note and/or prescription label, along with the item name, weight of the patient, and the number of units dispensed as input. A prompt for the element being classified is included. No examples are provided.

Few-shot Random Examples: Incorporates randomly-sampled example prescriptions or examination text and inferred labels from the training set as in-context examples (Brown et al., 2020). To manage token limits, we include three labeled examples for prescription prompts and two for indication prompts—the examination text required for the indication prompt were much longer.

Few-shot Similar Examples: Instead of random examples, we use text similarity as a selection criterion to retrieve examples for in-context learning (Zhang et al., 2023; Shi et al., 2023; Lewis et al., 2021). For the retriever, we employ a distilled `SBERT` model (Wang et al., 2020) to encode text and retrieve examples based on cosine similarity.

Functional Prompting: In the zero-shot setting, we leverage functional prompting (OpenAI, 2023b) with `GPT-3.5` and `GPT-4` to combine probabilistic outputs with rule-based calculations, enforcing data types for extracted prescription attributes and executing formulaic calculations for Dose and Duration, as detailed in §2. We compare these results with `LLAMA2-70B`'s configurations for extracting dose unit size and frequency. Additionally, we finetune a `VetBERT` model (Hur et al., 2020) with silver label data to isolate these attributes, and perform deterministic calculations for dose and duration.

Prompt Tuning To improve the performance of calculating the dose and duration of therapy, we include the formulas for the dose and duration calculations as part of the prompt. The prompts used for evaluation were additionally optimized using the framework proposed by Yang et al. (2023) to iteratively generate a set of prompts using `GPT-4`, test those prompts on a subset of records from the training set until no improvements were observed after multiple iterations, and keep the prompt with highest accuracy for each element. Final prompts can be found in Appendix A.2.

Postprocessing We remove non-numerical text and retain the first float in the model's output for enhanced accuracy, except for indication and ingredient which are expected to be strings.

	Ingredient	Indication	Dose	Duration	Frequency	Dose Unit Size
RxVetBERT	100	80.0	89.1	88.0	97.0	89.0
Few-Shot Similar Examples						
GPT-3.5	97.5	56.5	29.5	70.0	98.0	98.0
GPT-4	99.5	75.0	85.0	91.0	98.5	99.5
LLAMA2-70B	94.0	9.0	12.5	58.0	97.5	95.5
Few-Shot Random Examples						
GPT-3.5	67.0	73.5	26.0	61.0	98.5	97.0
GPT-4	100	73.5	88.5	84.5	98.5	100
LLAMA2-70B	42.0	27.5	9.5	61.0	97.5	92.5
Zero-Shot						
GPT-3.5	80.5	35.0	3.5	52.5	12.0	21.0
GPT-4	97.5	69.5	24.0	75.5	97.5	55.0
LLAMA2-70B	21.0	0.0	5.0	57.5	98.0	59.5

Table 1: Model accuracy (%) across multiple settings, benchmarked against RxVetBERT.

	Dose	Duration	Freq.	Dose Unit Size
Finetuned				
VetBERT	90.0	88.0	97.0	90.5
Few-Shot Similar Examples				
LLAMA2-70B	95.5	93.5	97.5	95.5
Zero-Shot				
GPT-3.5	94.5	92.5	98.0	98.0
GPT-4	99.5	98.0	98.5	99.5

Table 2: Evaluation of GPT-3.5 and GPT-4 in a zero-shot setting using functional prompts to enforce numerical data types, compared to VetBERT trained on silver labels and LLAMA2-70B in the Few-Shot Similar setting. Dose and Duration are computed deterministically for all model variants.

4 Results and Discussion

Overall, our experiments find that LLMs are highly effective at extracting and interpreting numerical elements (e.g., Frequency and Dose Unit Size) necessary to calculate the dose and duration (Table 1). GPT-3.5 and GPT-4 show high accuracy in the zero-shot setting and LLAMA2-70B using in-context learning examples. For all models, integrating probabilistic outputs with deterministic calculations through functional prompting achieves much higher accuracy for dose and duration values compared to directly prompting models for these values (Table 2). Functional prompting provides an effective way to ensure more reliable outcomes in tasks requiring numerical computations when those computations can be explicitly described.

The much smaller finetuned VetBERT achieves modest results, similar to silver label accuracy for dose unit size and frequency of administration, which suggests it could achieve higher accuracy with more accurate training labels. For active ingredient, we find LLMs to be highly effective with appropriate prompt tuning. Error analysis of initial results found that there were many errors for multi-ingredient medications, e.g., Amoxycillin Clavulanate incorrectly identified as Amoxycillin. Through prompt tuning, we identified the most effective way to overcome this as including the following prompt text: “focus on the active ingredients and note them all if they were present.”

Nonetheless, accurately pinpointing the primary indication for antimicrobial administration continues to be a challenging task, as evidenced by the lower inter-annotator agreement score for indications as discussed in §2, and the relatively poor performance of LLMs on this subtask. A closer examination of the errors reveals that they largely occurred in instances where the indication is ambiguous, similar to the complications noted earlier. Refining the labeling schema for indications is a promising avenue for mitigating this issue.

Conclusion This paper provides a framework for LLMs to extract essential prescription data from veterinary text, such as dose, duration, and active ingredients for supporting AMS efforts. We overcome limitations in calculating elements such as dose and duration by integrating probabilistic outputs with deterministic calculations through functional prompting, even in zero-shot settings. Future work should consider evaluation for human clinical applications, given the potential contributions of this approach to broader healthcare.

Limitations

The efficacy of in-context learning for models in the few-shot similar setting may be constrained by the precision of RxVetBERT, which was employed to furnish the examples used in the prompts. Random sampling was used to create a test set mirroring the full dataset population; this limited the diversity of specific disease syndromes in the test data, and may not provide a complete assessment of the models' capabilities.

LLMs are still prone to errors, even though they demonstrate high performance on evaluation datasets. When using an LLM for clinical decisions, it is critical that the final decisions involve clinicians as LLMs in their current state may still fall short. While our framework excels in identifying active ingredients, it faces challenges in ascertaining exact indications for medication, a more subjective task, signaling a potential direction for future work.

Acknowledgements

This work was supported in part by the National Institutes of Health, National Library of Medicine (NLM) University of Washington Biomedical Informatics and Data Science Research Training Program (Grant Nr LM 007442). The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. This research was undertaken with the assistance of information and other resources from the VetCompass Australia consortium under the project "VetCompass Australia: Big Data and Real-time Surveillance for Veterinary Science", which is supported by the Australian Government through the Australian Research Council LIEF scheme (LE160100026).

References

David Brodbelt. 2019. VeNom Coding – VeNom Coding Group.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, and Yi Zhang. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4.

Kevin J. Cummings, Victor A. Aprea, and Craig Altier. 2015. Antimicrobial resistance trends among canine Escherichia coli isolates obtained from clinical samples in the northeastern USA, 2004-2011. *The Canadian Veterinary Journal = La Revue Veterinaire Canadienne*, 56(4):393–398.

Peter Davey, Charis A Marwick, Claire L Scott, Esmita Charani, Kirsty McNeil, Erwin Brown, Ian M Gould, Craig R Ramsay, and Susan Michie. 2017. Interventions to improve antibiotic prescribing practices for hospital inpatients. *The Cochrane Database of Systematic Reviews*, 2017(2):CD003543.

John E. Ekakoro, G. Kenitra Hendrix, Lynn F. Guptill, and Audrey Ruple. 2022. Antimicrobial susceptibility and risk factors for resistance among Escherichia coli isolated from canine specimens submitted to a diagnostic laboratory in Indiana, 2010-2019. *PloS One*, 17(8):e0263949.

Marie F Grill and Rama K Maganti. 2011. Neurotoxic effects associated with antibiotic use: management considerations. *British Journal of Clinical Pharmacology*, 72(3):381–393.

Luca Guardabassi, Stefan Schwarz, and David H. Lloyd. 2004. Pet animals as reservoirs of antimicrobial-resistant bacteriaReview. *Journal of Antimicrobial Chemotherapy*, 54(2):321–332.

L. Y. Hardefeldt, J. R. Gilkerson, H. Billman-Jacobe, M. A. Stevenson, K. Thursky, G. F. Browning, and K. E. Bailey. 2018a. Antimicrobial labelling in Australia: a threat to antimicrobial stewardship? *Australian Veterinary Journal*, 96(5):151–154. Https://doi.org/10.1111/avj.12677.

L. Y. Hardefeldt, B. Hur, S. Richards, R. Scarborough, G. F. Browning, H. Billman-Jacobe, J. R. Gilkerson, J. Ierardo, M. Awad, R. Chay, and K. E. Bailey. 2022. Antimicrobial stewardship in companion animal practice: an implementation trial in 135 general practice veterinary clinics. *JAC-Antimicrobial Resistance*, 4(1):dlac015.

Laura Y. Hardefeldt, J. R. Gilkerson, H. Billman-Jacobe, M. A. Stevenson, K. Thursky, K. E. Bailey, and G. F. Browning. 2018b. Barriers to and enablers of implementing antimicrobial stewardship programs in veterinary practices. *Journal of Veterinary Internal Medicine*, 32(3):1092–1099.

B. Hur, L. Y. Hardefeldt, K. Verspoor, T. Baldwin, and J. R. Gilkerson. 2019. Using natural language processing and VetCompass to understand antimicrobial usage patterns in Australia. *Australian Veterinary Journal*, 97(8):298–300.

Brian Hur, Timothy Baldwin, Karin Verspoor, Laura Hardefeldt, and James Gilkerson. 2020. Domain Adaptation and Instance Selection for Disease Syndrome Classification over Veterinary Clinical Notes. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, pages 156–166, Online. Association for Computational Linguistics.

Brian Hur, Laura Y. Hardefeldt, Karin M. Verspoor, Timothy Baldwin, and James R. Gilkerson. 2022. Evaluating the dose, indication and agreement with guidelines of antimicrobial use in companion animal practice with natural language processing. *JAC-Antimicrobial Resistance*, 4(1):dlab194.

George Karystianis, Therese Sheppard, William G. Dixon, and Goran Nenadic. 2016. Modelling and extraction of variability in free-text medication prescriptions from an anonymised primary care electronic medical record research database. *BMC Medical Informatics and Decision Making*, 16.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. ArXiv:2005.11401 [cs].

Paul McGreevy, Peter Thomson, Navneet K. Dhand, David Raubenheimer, Sophie Masters, Caroline S. Mansfield, Timothy Baldwin, Ricardo J. Soares Magalhaes, Jacquie Rand, Peter Hill, Anne Peaston, James Gilkerson, Martin Combs, Shane Raidal, Peter Irwin, Peter Irons, Richard Squires, David Brodbelt, and Jeremy Hammond. 2017. VetCompass Australia: A National Big Data Collection System for Veterinary Science. *Animals : an Open Access Journal from MDPI*, 7(10):74.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023. Capabilities of GPT-4 on Medical Challenge Problems. ArXiv:2303.13375 [cs].

Jim O'Neill. 2016. Tackling drug-resistant infections globally: final report and recommendations. Report, Government of the United Kingdom.

OpenAI. 2023a. GPT-4 Technical Report. ArXiv:2303.08774 [cs].

OpenAI. 2023b. OpenAI Platform - Function Calling.

Dan G. O'Neill, Alison M. Skipper, Jade Kadhim, David B. Church, Dave C. Brodbelt, and Rowena M. A. Packer. 2019. Disorders of Bulldogs under primary veterinary care in the UK in 2013. *PLOS ONE*, 14(6):e0217928.

Jada L. Roe, Joseph M. Fuentes, and Michael E. Mullins. 2012. Underdosing of common antibiotics for obese patients in the ED. *The American Journal of Emergency Medicine*, 30(7):1212–1214.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2023. REPLUG: Retrieval-Augmented Black-Box Language Models. ArXiv:2301.12652 [cs].

Carson Tao, Michele Filannino, and Özlem Uzuner. 2017. Prescription extraction using CRFs and word embeddings. *Journal of Biomedical Informatics*, 72:60–66.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. ArXiv:2307.09288 [cs].

Özlem Uzuner, Imre Solti, Fei Xia, and Eithon Cadag. 2010. Community annotation experiment for ground truth generation for the i2b2 medication challenge. *Journal of the American Medical Informatics Association : JAMIA*, 17(5):519–523.

Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association : JAMIA*, 18(5):552–556.

Scott A. Waldman, Andre Terzic, Laurence J. Egan, Jean Luc Elghozi, Arshad Jahangir, Garvan C. Kane, Walter K. Kraft, Lionel D. Lewis, Jason D. Morrow, Leonid V. Zingman, Darrell R. Abernethy, Arthur J. Atkinson, Neal L. Benowitz, D. Craig Brater, Jean Gray, Peter K. Honig, Gregory L. Kearns, Barbara A. Levey, Stephen P. Spielberg, Richard Weinshilboum, and Raymond L. Woosley. 2008. *Pharmacology and Therapeutics: Principles to Practice*. Elsevier.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. MiniLM: Deep Self-Attention Distillation for Task-Agnostic

Compression of Pre-Trained Transformers. ArXiv:2002.10957 [cs].

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. 2023. Large Language Models as Optimizers. ArXiv:2309.03409 [cs].

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, and Songfang Huang. 2023. How well do Large Language Models perform in Arithmetic tasks? ArXiv:2304.02015 [cs].

Xuchao Zhang, Menglin Xia, Camille Couturier, Guoqing Zheng, Saravan Rajmohan, and Victor Ruhle. 2023. Hybrid Retrieval-Augmented Generation for Real-time Composition Assistance. ArXiv:2308.04215 [cs].

Chuanyang Zheng, Zhengying Liu, Enze Xie, Zhenguo Li, and Yu Li. 2023. Progressive-Hint Prompting Improves Reasoning in Large Language Models. ArXiv:2304.09797 [cs].

A Appendix

A.1 Prompt Design Details

This section details the prompt template used and provides an example prompt. Each initial prompt was generated using GPT-4 by prompting the model to generate 10 additional prompts for accomplishing the task. Each task was tested using GPT-3.5 and the process was repeated until the additional prompts were no longer improving the performance after 3 successive iterations. For dose and duration, which also required the steps for performing arithmetic, the formula for the necessary arithmetic was provided.

The prompt templates were designed as follows:

```
{PROMPT}

{Examples} (omitted in zero-shot settings)

{Instance to label}
```

Here is an example prompt for dose under the few-shot random setting:

```
Output the dosage in mg/kg. Dose is determined
by multiplying the total dose units given per
administration multiplied by the size of the
medication in mg, dividing by the weight of the
patient in kg to determine the mg per kg

** Example:
** Item Label: Disp By: ***: Dog 21.00 x Clinacin
   Tabs 150Mg One \& half (1.5) tablets twice a
   day with food ***
** Item Name: Clinacin Tabs 150Mg (100) Clindamycin
** Weight: 30kgs
** Medication Unit Size: 150.0
** Units Dispensed: 21.0
** Dose: 7.5

** Example:
** Item Label: PM:Disp:***: Cat 7.00 x Baytril 50Mg
   Tab Half (1/2) tablet once a day Give until
   finished. ***
** Item Name: Baytril 50Mg Tab (100) (enrofloxacin)
** Weight: 5kgs
** Medication Unit Size: 50.0
** Units Dispensed: 7.0
** Dose: 5

** Example:
** Item Label: Vet: ***: *** : Dog 10.00 x Veraflox
   Dog 60mg Give ONE (1) tablet ONCE a day Give
   until finished; ***
** Item Name: Veraflox Dog 60mg (70) (pradofloxacin)
** Weight: 30kgs
** Medication Unit Size: 60.0
** Units Dispensed: 10.0
** Dose: 2

** Instance to Label:
** Item Label: Vet: ***: Dog 10.00 x Clavulox Tabs
   500Mg One (1) tablet twice a day Give until
   finished. with food ***
** Item Name: Clavulox (Clavulanic Acid) Tabs 500Mg
   (100) **\\ Weight: 29kgs
** Medication Unit Size: 500.0
** Units Dispensed: 10.0
** Dose:
```

A.2 Prompts Used

This section details the example prompts used for each label in the task.

- **Active Ingredient:** “Referencing the trade name, choose the active ingredient from the Ingredients List that forms the medication. For combination drugs, ensure to select the ingredient with all components.”
- **Clinical Indication:** “Using the provided list of possible indications, give the most likely indication for the antimicrobial administration. If unclear from the text, label as unknown.”
- **Frequency:** “How many times per day is the medication given?”
- **Medication Size:** “What is the medication unit size in mg?”
- **Dosage Unit Size:** “How many units of the medication are given per dose?”
- **Overall Dose:** “Output the dosage in mg/kg. Dose is determined by multiplying the total dose units given per administration multiplied by the size of the tablet in mg, dividing by the

	Ingredient	*Indication*	*Dose*	*Duration*	*Frequency*	*Dose Unit Size*	*Weight*	*Total Units*	*Medication Size*
RxVetBERT	100	80.0	89.1	88.0	97.0	89.0	-	-	-
Few-Shot Similar Examples									
GPT-3.5	97.5	56.5	29.5	70.0	98.0	98.0	90.5	100	100
GPT-4	99.5	75.0	85.0	91.0	98.5	99.5	100	99.5	100
LLAMA2-70B	94.0	9.0	12.5	58.0	97.5	95.5	100	100	100
Few-Shot Random Examples									
GPT-3.5	67.0	73.5	26.0	61.0	98.5	97.0	99.5	100	100
GPT-4	100	73.5	88.5	84.5	98.5	100	100	99.5	100
LLAMA2-70B	42.0	27.5	9.5	61.0	97.5	92.5	100	100	100
Zero-Shot									
GPT-3.5	80.5	35.0	3.5	52.5	12.0	21.0	69.5	94	100
GPT-4	97.5	69.5	24.0	75.5	97.5	55.0	98.5	100	100
LLAMA2-70B	21.0	0.0	5.0	57.5	98.0	59.5	92.0	99.5	100

Table 3: Accuracy (%) of Large Language Models (LLMs) across multiple settings for all prescription elements, benchmarked against the RxVetBERT baseline ensemble methods.

weight of the patient in kg to determine the mg per kg."

- **Treatment Duration:** "Calculate the length of administration (in days) for the given prescription. To determine the length of administration, find the total number of tablets or doses dispensed and divide by the number of doses given per day."

A.3 Additional Evaluations

While evaluations were performed on all aspects of the prescription text, we omitted the performance of the elements which could be extracted directly out of the text, which were required for the dose calculations but did not require any conversion into numerical values, this included the Weight, Total Units, or Medication Size. We have included this in Table 3.

MiDRED: An Annotated Corpus for Microbiome Knowledge Base Construction

William Hogan[1], **Andrew Bartko**[2,3,4], **Jingbo Shang**[1], **Chun-Nan Hsu**[5]
[1]Department of Computer Science & Engineering,
[2]Center for Microbiome Innovation, [3]Department of Bioengineering,
[4]Department of Pediatrics, [5]Department of Neurosciences
University of California, San Diego, La Jolla, CA 92093
whogan@ucsd.edu

Abstract

The interplay between microbiota and diseases has emerged as a significant area of research facilitated by the proliferation of cost-effective and precise sequencing technologies. To keep track of the many findings, domain experts manually review publications to extract reported microbe-disease associations and compile them into knowledge bases. However, manual curation efforts struggle to keep up with the pace of publications. Relation extraction has demonstrated remarkable success in other domains, yet the availability of datasets supporting such methods within the domain of microbiome research remains limited. To bridge this gap, we introduce the Microbe-Disease Relation Extraction Dataset (MiDRED); a human-annotated dataset containing 3,116 annotations of fine-grained relationships between microbes and diseases. We hope this dataset will help address the scarcity of data in this crucial domain and facilitate the development of advanced text-mining solutions to automate the creation and maintenance of microbiome knowledge bases.

1 Introduction

Microbiota play a pivotal role in human health in diverse environments such as the gut, skin, and oral cavity, influencing various physiological processes and disease mechanisms (Cho and Blaser, 2012; Lynch and Pedersen, 2016; Singh et al., 2017). The significance of microbiome research is underscored by its immense potential to unlock new understandings and treatments for various health conditions (Stefano et al., 2022; Yu et al., 2022; Kustrimovic et al., 2023). For example, perturbations in gut microbiota composition, exemplified by fluctuations in Bacteroidetes and Firmicutes populations, have been linked to obesity and type 2 diabetes, respectively, providing valuable insights into the pathophysiology of these conditions (Baek et al., 2023; Kusnadi et al., 2023). The growth of microbiome research introduces significant challenges in knowledge consolidation and utilization (Badal et al., 2019; Huang et al., 2022). Current efforts often involve domain experts spending countless hours manually curating experimentally validated associations between diverse microbiota and diseases to form knowledge bases (KBs) (Li et al., 2021; Dai et al., 2021; Qi et al., 2022; Zhang et al., 2022). These KBs are invaluable for researchers and practitioners, providing a consolidated view of current findings, yet their maintenance is becoming unsustainable due to the rapid pace of publication. Advanced text-mining methods designed to extract knowledge from biomedical texts are a well-established area of research (Wei et al., 2016; Zhang et al., 2018; Hogan et al., 2021; Xu et al., 2022; Li, 2022; Lai et al., 2023; Liu et al., 2023). Methods often leverage human-annotated data to train and validate a model's performance; however, robust datasets annotating microbe-disease associations are lacking.

To address these challenges, we introduce MiDRED; a comprehensive text-mining dataset designed to automate the construction and maintenance of microbiome KBs. MiDRED consists of 3,116 annotated relationships between microbe-disease pairs extracted from 1,655 scholarly articles. We specifically craft relation classes to align with classes used in major microbe-disease KBs to ensure MiDRED's compatibility with existing databases. Importantly, MiDRED annotates negative instances (e.g., a "no relation" class) to mitigate positive bias from trained models (Zhang et al., 2017). MiDRED also includes span-level annotations of entities, which are crucial for training in Named Entity Recognition (NER) and Named Entity Normalization (NEN) tasks. See Table 2 for statistics on the complete dataset. We conducted experiments on MiDRED using a variety of generative and discriminative large language models to obtain robust baselines to serve as a foundation for future research. We openly release the MiDRED

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 398–408
August 16, 2024.

dataset on Hugging Face.[1]

2 Related Work

MiDRED is designed as a text-mining dataset and draws inspiration from numerous biomedical (Khettari et al., 2023; Bossy et al., 2019; Luo et al., 2022; Li et al., 2016; Taboureau et al., 2010; Janssens et al., 2018) and general domain text-mining datasets (Zhang et al., 2017; Stoica et al., 2021; Yao et al., 2019). Text-mining datasets typically consist of manually annotated texts which can be used to train and evaluate automated NER, NEN, and relation extraction algorithms (Zhang et al., 2017; Yao et al., 2019). Works such as Herrero-Zazo et al. (2013), Luo et al. (2022), González et al. (2019) are similar in task but differ either in entity types, association types, or both.

The Human Microbe-disease Dataset (HMDAD) (Ma et al., 2016) is a database of associations between human microbes and diseases. However, the dataset does not provide span-level information denoting entity pairs which limits the dataset's use in training NER and NEN algorithms. Microbes in HMDAD were primarily curated at the genus level due to the sequencing technologies available when the dataset was annotated. MiDRED benefits from advancements in sequencing technologies, allowing for a majority (95.4%) of microbial concepts to be annotated at the species level. Furthermore, HMDAD relation annotations are done at the article-level. Article-level annotation is commonly used in microbiome knowledge bases (Janssens et al., 2018; Cheng et al., 2019; Li et al., 2021; Skoufos et al., 2020) and fails to denote the location of textual evidence supporting an association, making it challenging to train automated text-mining tools. Lastly, MiDRED differs from HMDAD in that it does not limit its annotations based on host type, leading to more diverse associations.

The Species-species Interaction (SSI) dataset (Khettari et al., 2023) is a dataset that annotates binary associations between species of microbes. SSI does not provide human-annotated entities and relies on automated methods for NER. MiDRED differs from SSI in entity types and the number of relation classes—in MiDRED, we annotate four relation classes (see Section 3.2 for more details), moving beyond binary associations. Bacteria Biotope (BB 2019) (Bossy et al., 2019) is an NER/RE dataset featuring microbes, diseases, habitats, and locations. BB 2019 seeks to mine associations of microbes and environments (habitats) to better understand how microbes interact within various environments. MiDRED, in contrast, focuses on how microbes relate to diseases more generally and offers a large number of annotated entities and relations.

3 Methods

3.1 Data Collection and Entity Normalization

We collect an initial set of abstracts from PubMed (Sayers et al., 2020) using the PubTator tool (Wei). To ensure the subset of abstracts are relevant to microbiome studies, we prioritize PMIDs found within the Disbiome database (Janssens et al., 2018). From this subset, we randomly select abstracts and annotate microbes and diseases. Microbial entities were normalized to the List of Prokaryotic Names with Standing in Nomenclature (LPSN) ontology (Parte et al., 2020). Disease entities were normalized to the Comparative Toxicogenomics Database (CTD)(Davis et al., 2020). See Appendix A.1 for details about our entity annotation process.

3.2 Relation Annotation

As stated in Section 1, a primary goal of MiDRED is compatibility with existing microbiome KBs. As such, we align our relation classes to those used by major microbiome KBs and annotate four classes: *connecting*, *contrasting*, *pathogen*, and *no relation*. The *connecting* class aligns with positive classes (e.g., "associated," "increase," and "positive"), signifying a microbe is associated with a disease, while the *contrasting* class signifies a microbe that contrasts with a disease, aligning definitionally to negative classes (e.g., "reduce," "decrease," and "inhibit") (Qi et al., 2022; Janssens et al., 2018; Li et al., 2021; Zhang et al., 2022; Dai et al., 2021). We also include *pathogen*, which is a stronger, more causal relation compared to *connecting*, as well as *no relation* to help prevent positive bias. Each instance is double annotated by different annotators and conflicting annotations are resolved in a third annotation round. With this systematic approach, we achieved a high inter-annotator agreement (Fleiss' Kappa) of 0.710. See Appendix A.1.1 for additional details about the relation annotation process, class definitions, and examples (Table 7).

MiDRED's data splits are constructed by collecting the set of unique fact-triples (e.g., $\langle$*head*

[1] https://huggingface.co/datasets/shangdatalab-ucsd/midred

Dataset	Entity Types	Relation Classes	Host Type	Negative Instances	Entity Spans	# Microbes	# Relation Instances
HMDAD (Ma et al., 2016)	Microbes/diseases	2	Human	✗	✗	292	483
SSI (Khettari et al., 2023)	Microbes/microbes	2	Human	✔	✔	N/A*	999
Bacteria Biotope (Bossy et al., 2019)	Microbes/diseases/habitats/locs	2	Varied	✗	✔	1,760	2,639
MiDRED	Microbes/diseases	4	Varied	✔	✔	5,590	3,116

Table 1: A comparison between our proposed dataset, MiDRED, and other microbiome text-mining datasets. MiDRED features a multi-class relation classification task with annotated negatives (the "no relation" class) and span-level entity annotations (*the SSI dataset does not provide manually annotated entities).

entity, *relation*, *tail entity*⟩). Unique triples are divided into train, development, and test splits using 0.8/0.1/0.1 ratios, resulting in no overlapping fact-triples between data splits. See Appendix A.2 for statistics on each data split.

Documents:		1,655
Entities:	All	12,027 (678)
	Microbes	5,590 (197)
	Diseases	6,437 (482)
Relationships:	All	3,116
	Connecting	1,744
	Contrasting	161
	Pathogen	920
	No relation	291

Table 2: Counts of annotated entities and relationships in MiDRED. Parenthesized values denote the number of unique concepts. For detailed statistics on train, development, and test splits, see Appendix A.2.

4 Baseline Experiments

We explore the performance of popular NLP models using MiDRED on Named Entity Recognition (NER) and Relation Extraction (RE) tasks to establish the baseline performance and highlight challenging areas for future development.

4.1 Named Entity Recognition

In our NER experiments, we treat each entity mention span individually. We tested three NER models on our corpus: BiLSTM-CRF (Hochreiter and Schmidhuber, 1997), BioBERT-CRF (Lee et al., 2019), and PubMedBERT-CRF (Gu et al., 2020). Sentences were transformed into hidden state vector sequences by the respective models. Each model was tasked with predicting the labels for each token within these sequences. Subsequently, a fully connected layer was employed to calculate the network score, and a conditional random field (CRF) layer decoded the optimal tag path from all possible paths, utilizing the BIO (Begin, Inside, Outside) tagging scheme to categorize each token accurately. See Appendix A.4 for hyperparameter details.

Model	P	R	F1
BiLSTM-CRF	0.877	0.891	0.884
PubMedBERT-CRF	0.947	0.972	0.959
BioBERT-CRF	**0.957**	**0.981**	**0.969**

Table 3: Precision, recall, and F1-micro scores of various NER models on the MiDRED test set. Results are averages from three runs.

4.2 Relation Extraction

For RE experiments, we explore fine-tuning encoder-only biomedical language models (BioLinkBERT (Yasunaga et al., 2022) and PubMedBERT (Gu et al., 2020)). We send representations for the [CLS] token through a fully connected layer trained with cross-entropy. Additionally, we explore the current in-context learning abilities of frontier LLMs (GPT 3.5 (OpenAI, 2021) and GPT 4 (OpenAI et al., 2024))[2]. For details on the prompt we use, see Appendix A.5.

5 Results and Discussion

Figure 1 displays the top ten microbes and diseases and the distribution of relation classes in MiDRED. We observe a long-tail distribution for both entity types. The distribution of microbes, in particular, features a steep drop-off in mention frequency after the most mentioned microbe, Helicobacter pylori, indicating that current research focuses on a relatively narrow set of microbes.

We observe relatively high scores for both the NER (Table 3) and RE (Table 4) experiments when looking at performance across all test instances using small, fine-tuned biomedical language models (PubMedBERT$_{\text{base}}$ and BioLinkBERT$_{\text{large}}$), indicating the effectiveness of modern information ex-

[2] Specifically, we use *gpt-3.5-turbo-16k-0613* and *gpt-4-turbo-preview* via OpenAI's API, accessed on 5/3/2024.

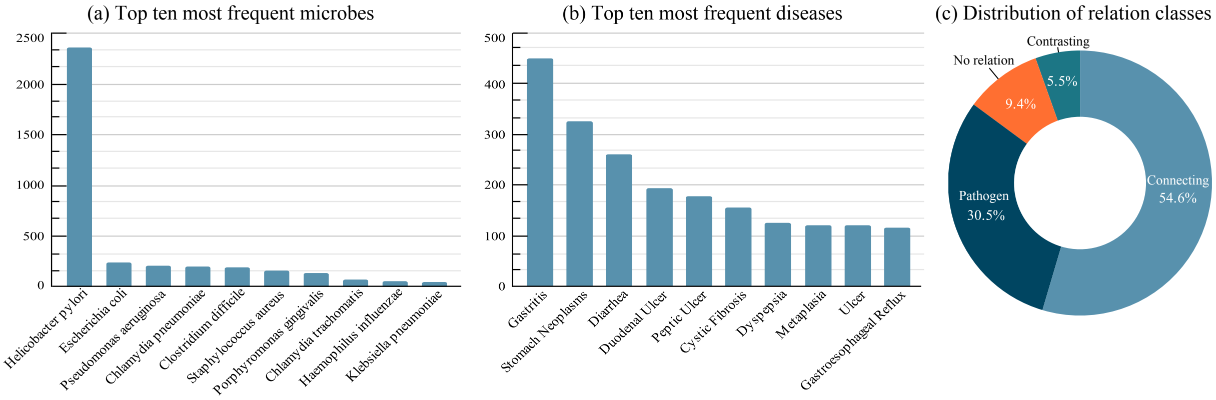


Figure 1: Counts of the top ten most frequent (a) microbial and (b) disease concepts, as well as (c) the distribution of relation classes found in the combined splits of MiDRED.

Model	**P**	**R**	**F1**
PubMedBERT$_{\text{base}}$	0.867	0.855	0.861
BioLinkBERT$_{\text{large}}$	**0.907**	**0.904**	**0.905**
GPT 3.5	0.542	0.562	0.552
GPT 4	0.716	0.725	0.721

Table 4: Precision, recall, and F1-micro scores of relation extraction models on the test set.

Model	**Q1**	**Q2**	**Q3**	**Q4**
PubMedBERT$_{\text{base}}$	0.895	0.852	0.800	0.571
BioLinkBERT$_{\text{large}}$	**0.929**	**0.839**	**0.801**	0.601
GPT 3.5	0.512	0.533	0.402	0.600
GPT 4	0.696	0.710	0.606	**0.667**

Table 5: F1-micro scores of RE models on test instances decomposed into quartiles based on microbe frequency, where Q1 is the performance on triples containing the top 25% most frequent microbes across all of MiDRED, followed by Q2, Q3, and finally, the least frequent quartile of microbes in Q4.

traction methods. Large, general domain language models (GPT 3.5 and GPT 4) leveraging in-context learning struggle to identify relations compared to smaller biomedical language models. This aligns with Peng et al. (2024)'s findings, offering additional evidence that large language models have yet to overtake smaller language models in information extraction tasks.

Furthermore, Table 5 shows a steady drop-off in PubMedBERT$_{\text{base}}$ and BioLinkBERT$_{\text{large}}$'s performance across quartiles of test triples decomposed based on microbe frequency, while the performance of GPT 3.5 and 4 remains relatively stable. This indicates that the smaller models generalize poorly and signify an area for future development.

Annotation Challenges: Numerous challenges were encountered when annotating microbes, diseases, and their associations. Challenges with acronyms and abbreviations arose due to variations in naming conventions, which sometimes differed from standard classifications. Relation types posed difficulties in accurately describing the links between microbe-disease pairs, particularly in cases involving numerical data or complex biological semantics. We record these and other challenges in Appendix A.3 in hopes of improving future versions of MiDRED and biomedical annotation efforts in general.

6 Conclusion

Microbiota, integral to human health and prevalent in various body environments like the gut, skin, and oral cavity, are at the forefront of promising research avenues that could revolutionize our understanding and treatment of numerous health conditions. However, the manual curation of microbiome knowledge bases, though invaluable, faces scalability challenges in keeping pace with the rapid influx of new research findings. In this paper, we introduce MiDRED, a dataset that aims to bridge this gap by providing a resource to help automate the creation and maintenance of microbiome bases. MiDRED can be used to train and validate state-of-the-art NLP models on various tasks such as named entity recognition, named entity normalization, bacteria-disease relationship extraction, and knowledge graph creation. We hope MiDRED will unlock new applications and innovations within microbiome research.

Limitations

MiDRED is a sentence-level annotated dataset, which inherently limits its scope to capturing relationships expressed within individual sentences. Consequently, the dataset does not encompass inter-sentence relationships, which could provide additional context and depth to understanding microbe-disease interactions. Furthermore, MiDRED maintains a focused thematic scope, exclusively concentrating on relationships between microbes and diseases. While beneficial for depth and specificity in this area, this focus excludes potential relationships involving other biological entities or environmental factors that could influence or be influenced by the microbe-disease dynamics. Such annotations could offer deeper insights into the context and contingencies of the documented relationships. We aim to address these limitations in future versions of the dataset.

Ethics Statement

In the development and release of the MiDRED dataset, we have carefully considered ethical aspects and do not anticipate any major ethical concerns. The dataset is constructed from publicly available academic articles, focusing solely on the relationships between microbes and diseases without involving individual patient data or personal information. By openly releasing the MiDRED dataset, we commit to facilitating transparency in our research process. This open access approach allows for peer review, replication of results, and collaborative improvements to the dataset.

Acknowledgements

Many thanks to Ho-Cheol Kim and Yannis Katsis for their support in all aspects of this project. Thank you to the anonymous reviewers for their thoughtful comments and corrections. This work is supported by IBM Research AI through the AI Horizons Network.

References

pubtator central: automated concept annotation for biomedical full text articles.

Varsha D. Badal, Dustin Wright, Yannis Katsis, Ho-Cheol Kim, Austin D. Swafford, Rob Knight, and Chun-Nan Hsu. 2019. Challenges in the construction of knowledge bases for human microbiome-disease associations. *Microbiome*, 7.

Ga Hyeon Baek, Ki-Myeong Yoo, Seon-Yeong Kim, Da Hee Lee, Hayoung Chung, Suk-Chae Jung, Sung-Kyun Park, and Jun-Seob Kim. 2023. Collagen Peptide Exerts an Anti-Obesity Effect by Influencing the Firmicutes/Bacteroidetes Ratio in the Gut. *Nutrients*, 15.

Robert Bossy, Louise Deléger, Estelle Chaix, Mouhamadou Ba, and Claire Nédellec. 2019. Bacteria Biotope at BioNLP Open Shared Tasks 2019. In *Conference on Empirical Methods in Natural Language Processing*.

Liang Cheng, Changlu Qi, Zhuang He, Tongze Fu, and Xue Zhang. 2019. gutMDisorder: a comprehensive database for dysbiosis of the gut microbiota in disorders and interventions. *Nucleic Acids Research*, 48:D554 – D560.

Ilseung Cho and Martin J. Blaser. 2012. The human microbiome: at the interface of health and disease. *Nature Reviews Genetics*, 13:260–270.

Die Dai, Jiaying Zhu, Chuqing Sun, Min Li, Jinxin Liu, Sicheng Wu, Kang Ning, Li-jie He, Xing-Ming Zhao, and Wei-Hua Chen. 2021. GMrepo v2: a curated human gut microbiome database with special focus on disease markers and cross-dataset comparison. *Nucleic Acids Research*, 50(D1):D777–D784.

Allan Peter Davis, Cynthia J. Grondin, Robin J. Johnson, Daniela Sciaky, Jolene Wiegers, Thomas C. Wiegers, and Carolyn J. Mattingly. 2020. Comparative Toxicogenomics Database (CTD): update 2021. *Nucleic Acids Research*, 49:D1138 – D1143.

Janet Piñero González, Juan Manuel Ramírez-Anguita, Josep Saüch-Pitarch, Francesco Ronzano, Emilio Centeno, Ferran Sanz, and Laura Inés Furlong. 2019. The DisGeNET knowledge platform for disease genomics: 2019 update. *Nucleic Acids Research*, 48:D845 – D855.

Yu Gu, Robert Tinn, Hao Cheng, Michael R. Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2020. Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3:1 – 23.

María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. 2013. The DDI corpus: An annotated corpus with pharmacological substances and drug-drug interactions. *Journal of biomedical informatics*, 46 5:914–20.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9:1735–1780.

William P Hogan, Molly Huang, Yannis Katsis, Tyler Baldwin, Ho-Cheol Kim, Yoshiki Baeza, Andrew Bartko, and Chun-Nan Hsu. 2021. Abstractified Multi-instance Learning (AMIL) for Biomedical Relation Extraction. In *3rd Conference on Automated Knowledge Base Construction*.

Zhiqiang Huang, Kun C. Liu, Wenwen Ma, Dezhi Li, Tianlu Mo, and Qing Liu. 2022. The gut microbiome in human health and disease—where are we and where are we going? a bibliometric analysis. *Frontiers in Microbiology*, 13.

Rezarta Islamaj, Dongseop Kwon, Sun Kim, and Zhiyong Lu. 2020. TeamTat: a collaborative text annotation tool.

Yorick Janssens, Joachim Nielandt, Antoon Bronselaer, Nathan Debunne, Frederick Verbeke, Evelien Wynendaele, Filip Van Immerseel, Yves-Paul Vandewynckel, Guy De Tré, and Bart de Spiegeleer. 2018. Disbiome database: linking the microbiome to disease. *BMC Microbiology*, 18.

Oumaima El Khettari, Solen Quiniou, and Samuel Chaffron. 2023. Building a Corpus for Biomedical Relation Extraction of Species Mentions. In *Workshop on Biomedical Natural Language Processing*.

Yulianto Kusnadi, Mgs Irsan Saleh, Zulkhair Ali, Hermansyah Hermansyah, Krisna Murti, Zen Hafy, and Eddy Yuristo. 2023. Firmicutes/Bacteroidetes Ratio of Gut Microbiota and Its Relationships with Clinical Parameters of Type 2 Diabetes Mellitus: A Systematic Review. *Open Access Macedonian Journal of Medical Sciences*.

Natasha Z. Kustrimovic, Raffaella Bombelli, Denisa Baci, and Lorenzo Mortara. 2023. Microbiome and Prostate Cancer: A Novel Target for Prevention and Treatment. *International Journal of Molecular Sciences*, 24.

Po-Ting Lai, Chih-Hsuan Wei, Ling Luo, Qingyu Chen, and Zhiyong Lu. 2023. BioREx: Improving Biomedical Relation Extraction by Leveraging Heterogeneous Datasets. *ArXiv*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36:1234 – 1240.

Jiacheng Li. 2022. SPOT: Knowledge-Enhanced Language Representations for Information Extraction. *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. 2016. BioCreative V CDR task corpus: a resource for chemical disease relation extraction. *Database: The Journal of Biological Databases and Curation*, 2016.

Longqing Li, Qingxu Jing, Sen Yan, Xuxu Liu, Yuanyuan Sun, Defu Zhu, Dawei Wang, Chenjun Hao, and Dongbo Xue. 2021. Amadis: A Comprehensive Database for Association Between Microbiota and Disease. *Frontiers in Physiology*, 12.

Haiyan Liu, Pingping Bing, Mei jun Zhang, Geng Tian, Jun Ma, Haigang Li, Meihua Bao, Kunhui He, Jianjun He, Binsheng He, and Jialiang Yang. 2023. MNNMDA: Predicting human microbe-disease association via a method to minimize matrix nuclear norm. *Computational and Structural Biotechnology Journal*, 21:1414 – 1423.

Ling Luo, Po-Ting Lai, Chih-Hsuan Wei, Cecilia Noemi Arighi, and Zhiyong Lu. 2022. BioRED: a rich biomedical relation extraction dataset. *Briefings in Bioinformatics*, 23.

Susan V. Lynch and Oluf Pedersen. 2016. The Human Intestinal Microbiome in Health and Disease. *The New England Journal of Medicine*, 375 24:2369–2379.

Wei Ma, Lu Zhang, Pan Zeng, Chuanbo Huang, Jianwei Li, Bin Geng, Jichun Yang, Wei Kong, Xuezhong Zhou, and Qinghua Cui. 2016. An analysis of human microbe–disease associations. *Briefings in Bioinformatics*, 18(1):85–97.

OpenAI. 2021. ChatGPT-3.5: Optimizing Language Models for Dialogue. Available: https://openai.com/blog/chatgpt-3-5/.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. GPT-4 Technical Report.

A.C. Parte, J. Sardà Carbasse, J.P. Meier-Kolthoff, L.C. Reimer, and M. Göker. 2020. List of Prokaryotic names with Standing in Nomenclature (LPSN).

Letian Peng, Zilong Wang, Feng Yao, Zihan Wang, and Jingbo Shang. 2024. MetaIE: Distilling a Meta Model from LLM for All Kinds of Information Extraction Tasks.

Changlu Qi, Yiting Cai, Kai Qian, Xuefeng Li, Jia-Yi Ren, Ping Wang, Tongze Fu, Tianyi Zhao, Liang Cheng, Lei Shi, and Xue Zhang. 2022. gutMDisorder v2.0: a comprehensive database for dysbiosis of gut microbiota in phenotypes and interventions. *Nucleic Acids Research*, 51:D717 – D722.

Eric W. Sayers, Jeff Beck, Evan E. Bolton, Devon Bourexis, James Rodney Brister, Kathi Canese, Donald C. Comeau, Kathryn Funk, Sunghwan Kim, William Klimke, Aron Marchler-Bauer, Melissa J. Landrum, Stacy Lathrop, Zhiyong Lu, Thomas L. Madden, Nuala A. O'Leary, Lon Phan, Sanjida H. Rangwala, Valerie A. Schneider, Yuri Skripchenko, Jiyao Wang, Jian Ye, Barton W. Trawick, Kim D. Pruitt, and Stephen T. Sherry. 2020. Database resources of the National Center for Biotechnology Information. *Nucleic acids research*.

Rasnik K. Singh, Hsin Wen Chang, Di Yan, Kristina M. Lee, Derya Uçmak, Kirsten Wong, Michael Abrouk, Benjamin Farahnik, Mio Nakamura, Tian Hao Zhu, Tina Bhutani, and Wilson J. Liao. 2017. Influence of diet on the gut microbiome and implications for human health. *Journal of Translational Medicine*, 15.

Giorgos Skoufos, Filippos S. Kardaras, Athanasios Alexiou, Ioannis Kavakiotis, Anastasia Lambropoulou, Vasiliki Kotsira, Spyros Tastsoglou, and Artemis G. Hatzigeorgiou. 2020. Peryton: a manual collection of experimentally supported microbe-disease associations. *Nucleic Acids Research*, 49:D1328 – D1333.

Mattia Di Stefano, Alessandro Polizzi, Simona Santonocito, Alessandra Romano, Teresa Lombardi, and Gaetano Isola. 2022. Impact of Oral Microbiome in Periodontal Health and Periodontitis: A Critical Review on Prevention and Treatment. *International Journal of Molecular Sciences*, 23.

George Stoica, Emmanouil Antonios Platanios, and Barnab'as P'oczos. 2021. Re-TACRED: Addressing Shortcomings of the TACRED Dataset. In *AAAI Conference on Artificial Intelligence*.

Olivier Taboureau, Sonny Kim Nielsen, Karine Audouze, Nils Weinhold, Daniel Edsgärd, Francisco S. Roque, Irene Kouskoumvekaki, Alina Bora, Ramona Curpan, Thomas Skøt Jensen, Søren Brunak, and Tudor I. Oprea. 2010. ChemProt: a disease chemical biology database. *Nucleic Acids Research*, 39:D367 – D372.

Xinyi Wang, Wanrong Zhu, and William Yang Wang. 2023. Large Language Models Are Implicitly Topic Models: Explaining and Finding Good Demonstrations for In-Context Learning.

Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Jiao Li, Thomas C. Wiegers, and Zhiyong Lu. 2016. Assessing the state of the art in biomedical relation extraction: overview of the BioCreative V chemical-disease relation (CDR) task. *Database: The Journal of Biological Databases and Curation*, 2016.

Jerry W. Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, and Tengyu Ma. 2023. Larger language models do in-context learning differently. *ArXiv*, abs/2303.03846.

Jiashu Xu, Mingyu Derek Ma, and Muhao Chen. 2022. Can NLI Provide Proper Indirect Supervision for Low-resource Biomedical Relation Extraction? *ArXiv*, abs/2212.10784.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. DocRED: A Large-Scale Document-Level Relation Extraction Dataset. *ArXiv*, abs/1906.06127.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2022. LinkBERT: Pretraining Language Models with Document Links. In *Annual Meeting of the Association for Computational Linguistics*.

Irene S. Yu, Rongrong Wu, Yoshihisa Tokumaru, Krista P. Terracina, and Kazuaki Takabe. 2022. The Role of the Microbiome on the Pathogenesis and Treatment of Colorectal Cancer. *Cancers*, 14.

J Zhang, Xiqian Chen, Jiaxin Zou, Chen Li, Wanying Kang, Yang Guo, Sheng Liu, Wenjing Zhao, Xiangyu Mou, Jiayuan Huang, and Jia Ke. 2022. MADET: a Manually Curated Knowledge Base for Microbiomic Effects on Efficacy and Toxicity of Anticancer Treatments. *Microbiology Spectrum*, 10.

Yijia Zhang, Hongfei Lin, Zhihao Yang, Jian Wang, Shaowu Zhang, Yuanyuan Sun, and Liang Yang. 2018. A hybrid model based on neural networks for biomedical relation extraction. *Journal of biomedical informatics*, 81:83–92.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware Attention and Supervised Data Improve Slot Filling. In *Conference on Empirical Methods in Natural Language Processing*.

A Appendix

A.1 Annotation Details

We developed an in-house annotation tool with functionality similar to open-source annotation tools such as Islamaj et al. (2020) to aid in annotating entities and relationships. When annotating articles, annotators can tag text and select “disease” or “microbe” based on the entity they intend to annotate. Depending on their choice, a select box displays a list of microbes from the List of Prokaryotic names with Standing in Nomenclature (LPSN) (Parte et al., 2020) dictionary or diseases based on the Comparative Toxicogenomics Database (CTD) (Davis et al., 2020) dictionary, allowing for the selection of an ontology concept. The annotation tool presents annotators with a list of potential microbe or disease concept matches sorted based on the mention text’s similarity to the concepts and concept synonyms in the corresponding ontology. Individual diseases and microbe concepts can also be searched for using quotes. The selection of either the disease or microbe allows for the normalization of entities.

Additionally, the annotation tool we developed has multiple features to aid the annotating process. It underlines the annotated text based on selected entities, with microbe entities underlined in purple and disease entities underlined in orange, allowing for quick verification by the annotator. Furthermore, annotators can quickly cycle through, delete, and clear annotations using select keys, decreasing the annotating process’s time-intensiveness.

Normalization is a classification process that classifies the different named entities of the same disease or microbe into a unique concept. Annotators were instructed to label microbes and diseases, including full names, abbreviations, synonyms, and acronyms. Adjectives and entities beyond LPSN and CTD databases were not annotated.

A.1.1 Annotating Relationships

After our entity annotation process, single sentences containing at least one microbe-disease pair were extracted and split into two subgroups. Sentences that had 80 characters or less and contained a rule-based keyword (Table 6) were placed into *Group A*, while all other sentences were placed into *Group B*. *Group A* sentences were then given pre-labels by rule-based algorithms (Table 6) concluded from observations in pilot annotation trials. Each assigned relation type was later manually verified by human annotators. Sentences in *Group B* were all manually labelled with relation types by human annotators. Each sentence across both groups were doubly-annotated to ensure the accuracy of the annotations. Instances of conflicting annotations were re-visited and relabeled in a third round of annotation. Using this process, we observe an inter-annotator agreement (Fleiss’ Kappa) of 0.710, indicating high annotator agreement.

In pilot observation trials, we found that in describing relationships in which the microbial entity favored the development of the disease entity, a positive relation type was insufficient to encompass all associations. Thus, we employed two positive relation types of *pathogen* and *connecting*. *Pathogen* is used for more explicitly defined cases, where the

Relation type	Rule-based Keywords
Connecting	Associated, antibody, initiate, increase, develop, positive, accelerate, triggered, recognized, identify, colonized, diagnose, eradication+decrease, isolate
Contrasting	Reduce, decrease, eradication+increase, inhibit+proliferation, induced+delayed, inhibit
Pathogen	Caused, pathogen, agent, induce, due to
No relation	Not associated with, not present in, no effect against

Table 6: Keywords for pre-labelling rules used for annotating relationships.

microbe is a pathogen or causative agent for the disease or characterizes a particular sub-type of the disease. *Connecting* is used when the microbe is associated with or is a risk factor for the disease. See Table 7 for definitions of each relation class.

A.2 Data Splits

As mentioned in Section A.1.1, MiDRED is split using a holdout set of fact triples. This ensures that trained models cannot simply memorize relationships between head and tail entities. In Table 8, we show the statistics of each data split in MiDRED.

A.3 Challenges

In this section, we openly discuss the challenges we faced in annotating microbe and disease entities and associating relations. We hope these lessons will inform subsequent versions of MiDRED and future biomedical annotation efforts.

A.3.1 Challenges with acronyms and abbreviations:

While microbial and disease entities in this dataset were fully normalized to respective classification standards, challenges and limitations were encountered during the annotation process. As a nomenclature convention, bacterium names are often abbreviated after the first introduction. As a result, bacteria mentions had to be normalized, with its abbreviated form, which could differ from paper to paper. Similar challenges were found in disease acronyms, while compounded and embedded naming involving disease acronyms brings extra complexity. Moreover, while bacteria mentions follow relatively rigid and uniform nomenclature standards, disease mentions are more flexible and versatile according to authors' naming and writing style. With the differing naming techniques of authors, disease and bacteria entities were occasionally not encompassed by LPSN or CTD dictionaries and, therefore, unable to be annotated. Unnormalized entities were excluded from MiDRED and thus could be missed in developing computational models.

A.3.2 Challenges with relation types:

We found that *pathogen*, *connecting*, *contrasting*, and *no relation* relation types could not describe the linking relation between all microbe-disease pairs. As mentioned in the Limitations section, annotation units were annotated in single sentences, which led to lost context and instances where we could not determine a relation type and associations. A similar problem occurred when numbers were involved, for instance:

> ***Helicobacter pylori*** was found in 12 of 13 **AIDP** patients (92%), and in 10 of 20 controls (50%), (P = 0.02). (PMID: 15679702)

Although the four relation types, particularly *connecting* and *contrasting*, could be inferred from cases in which numbers were involved, more often than not, we felt that the numbers were taken out of context, and perceived relation types could be inaccurate from just the sentence. Consequently, we decided not to label all cases in which numbers were needed to determine relation types.

As the proposed four relation types were used to successfully label most annotation units, two commonly encountered complexity issues need to be further addressed in future annotation efforts:

1. **Relations Dependent on Quantitative Semantics:** As *connecting* and *contracting* relation types categorize the directions of associated development, which are often hinted at by keywords, more specific descriptions of experiments are often presented in quantitative data. As a single sentence can only provide limited information, the implication of the quantities is sometimes indefinite, as in the following example:

   > During the study period, a total of 373 blood cultures were obtained from patients in whom **brucellosis** was suspected, and 27 (7.2%) of

Relation type	Definition	Example
Connecting	The microbe is a risk factor for the development of the disease.	BACKGROUND: The presence of ***Mycoplasma pneumoniae*** has been associated with worsening **asthma** in children.
	The microbe is associated with the disease.	The ***Helicobacter pylori*** (H. pylori) bacterium has been classified by the World Health Organization as a type 1 carcinogen with associations to the development of peptic and gastric ulcers, **gastric carcinoma** and primary B-cell lymphoma.
Contrasting	The microbe or substances extracted from it is beneficial for the treatment of the disease.	Bacille Calmette-Gurin (BCG), an attenuated strain of ***Mycobacterium bovis***, is one of the most effective agents in the treatment of **superficial bladder cancer**.
	The microbe is beneficial in the improvement of the disease.	CONCLUSION: ***Lactobacillus reuteri*** effectively reduced the duration of acute **diarrhea** and hospital stays in children hospitalised with acute gastroenteritis.
No relation	No association between disease and microbe.	A high density of ***H. pylori*** colonization in the gastric mucosa was not associated with a higher frequency of **dyspepsia** (P > 0.80).
Pathogen	The microbe is a pathogen/causative agent for the disease.	*Orientia tsutsugamushi* (***O. tsutsugamushi***), the causative agent of **scrub typhus**, is an obligate intracellular pathogen.
	The microbe name is used immediately preceding the disease name to form a specific subtype of the disease	Fifteen children (41%) had ulcers associated with ***H. pylori*** **gastritis**, including all 10 children with a chronic ulcer.

Table 7: Classification standards for microbe-disease relation annotation used when annotating MiDRED. Annotated microbe and disease concepts are in bold.

> them, drawn from 21 different patients, were positive for ***B. melitensis***. (PMID: 7989539)

2. **Relations Dependent on Biological Semantics:** The relations between microbe and host are essentially dynamic biological processes that, in many cases, can hardly be interpreted without implementing biological semantics. For instance, concepts such as vaccine, attenuated strains, microbe eradication, and co-infections are sometimes used in sentences, and excluding these semantics in the annotation process often leads to incorrect labels. Below is an example that our annotators found ambiguous without additional context from biological semantics:

> These results demonstrate that ***B. burgdorferi***-specific T lymphocytes primed by vaccination with a whole-cell preparation of inactivated B. burgdorferi sensu stricto isolate C-1-11 in adjuvant are involved in the development of **severe destructive arthritis**. (PMID: 7890402)

In the current version of MiDRED, such instances

Annotations		Train	Dev	Test	All
Documents		1,521	521	549	1,655
Entities	All	8,985 (613)	1,452 (311)	1,590 (310)	12,027 (678)
	Microbes	4,182 (179)	687 (100)	721 (95)	5,590 (197)
	Diseases	4,803 (435)	765 (212)	869 (216)	6,437 (482)
Relationships	All	2,169	447	500	3,116
	Connecting	1,224	248	272	1,744
	Contrasting	100	29	32	161
	Pathogen	635	132	153	920
	No relation	210	38	43	291

Table 8: Counts of entities and relationships annotated in the MiDRED dataset across the train, development, and test data splits. Parenthesized values denote counts of unique concepts.

are excluded from the dataset as they cast challenges for human annotators and the design of the classification standards. We intend to rectify these issues in future versions of the dataset.

A.3.3 Challenges with relation annotations:

There were some limitations to the rule-based prelabelling that we employed, as we could not assign rule-based pre-labels to *Group B* sentences. The reasoning behind this was twofold. *Group B* housed all the sentences without rule-based keywords (Table 6), so we could not give pre-labels by rule-based algorithms as we had done with *Group A*. Furthermore, *Group B* sentences were longer, and relations dependent on biological semantics were encountered more often, which required human annotators to interpret individual cases. Based on these challenges, we decided to forego rule-based pre-labeling on *Group B* sentences, resulting in these sentences being subject to more ambiguity.

A.4 Baseline NER Settings

For our NER experiments in 4, we use the following hyperparameter settings: 1,024 embedding dimensions, 512 max sequence length, and 64 batch size. We trained BioLinkBERT-CRF and PubMedBERT-CRF over three epochs and the BiLSTM-CRF for ten epochs.

A.5 GPT 3.5 and GPT 4 Prompts

GPT 3.5 and GPT 4 often perform better on tasks with the help of in-context learning (Wei et al., 2023; Wang et al., 2023). We construct a prompt that lists all relation classes and offers a couple of examples of extracted relationships. The following is the prompt we used for soliciting predictions for our tests:

> You are a relation extraction expert tasked with labeling relationships between head and tail entities in a sentence. Each example below has the head and tail entities appended to the sentence in the form: (head: head entity) (tail: tail entity). Predict if the sentence expresses one of the four following relation classes: “no relation”, “connecting”, “contrasting”, “pathogen”. The following are some examples:
>
> ### Sentence: At day 0 , 25 acute ulcers were associated with chronic H. pylori gastritis ; one patient had neither gastritis nor H. pylori infection (head: “H. pylori”) (tail: “ulcers”)
>
> ### Label: connecting
>
> . . . *[We include 4x examples of each relation class in the prompt.]* . . .
>
> ### Sentence: Significant resistance enhancement of mice pretreated with P. acnes against vaccinia virus or herpes simplex virus type 1 infection was observed. (head: “P. acnes”) (tail: “herpes simplex”)
>
> ### Label: ?

GPT 3.5 and GPT 4 responses were then aligned to ground truth classes via partial string matching for evaluation.

Do numbers matter? Types and prevalence of numbers in clinical texts

Rahmad Mahendra, Damiano Spina, Lawrence Cavedon, and **Karin Verspoor**
RMIT University
Melbourne, Australia
rahmad.mahendra@student.rmit.edu.au
{damiano.spina, lawrence.cavedon, karin.verspoor}@rmit.edu.au

Abstract

In this short position paper, we highlight the importance of numbers in clinical text. We first present a taxonomy of number variants. We then perform corpus analysis to analyze characteristics of number use in several clinical corpora. Based on our findings of extensive use of numbers, and limited understanding of the impact of numbers on clinical NLP tasks, we identify the need for a public benchmark that will support investigation of numerical processing tasks for the clinical domain.

1 Introduction

Numbers comprise a considerable amount of textual content and contribute substantially to conveying meaning in a range of domains including financial and scientific contexts. Targeted strategies for representing numbers have been shown to improve general literacy of language models (Thawani et al., 2021a). Numbers pose challenges for Natural Language Processing (NLP) due to their varied representations in text, as digits, words, or numerical expressions. This requires NLP models to handle ambiguity and context effectively in interpreting numerical information (Thawani et al., 2021b).

Numerical reasoning is crucial in the generative large language model (LLM) era because it underpins data-driven decision-making in many fields, including clinical, where accurate numerical insights are essential. LLMs often struggle with arithmetic operations and unit conversions, impacting their reliability in quantitative tasks. In the clinical domain, accurate numerical reasoning is vital for analyzing trial data, interpreting results, and making precise treatment recommendations, such as determining appropriate drug dosages based on statistical analyses of patient outcomes. For example, generating a report that states "The mean number of antihypertensive medication classes increased from 1.6 (95% CI, 1.4-1.8) at baseline to 2.2 (95% CI 2.0-2.4) at 6 months"[1] requires precise numerical reasoning.

In this paper, we characterize the numerical information in clinical NLP corpora. Through corpus analysis, we find that numerical information is frequent, but only a small portion is annotated or utilized. Our analysis identifies a number of issues concerning numeracy that need further attention from the clinical NLP research community.

2 Numerical strings and types

There is an assumption that numbers may be trivially extracted from text, as they consist of digit sequences or a finite set of numerals. However, numerical information can appear in various lexical surface and semantic contexts (Hanauer et al., 2019; Miok et al., 2023). We identify a multitude of number variants, and further define semantic categories for numerical information.

Numerical values can be expressed in many forms: (i) **Digit**, including integer ('3', '100,000'), float ('0.5'), and negative ('-1'), (ii) **Number with unit** ('100 mg', '160/90mmHg'), (iii) **Fraction**, written using the division symbol ('5/32') or with a special symbol (½), (iv) **Number range** ('from 0 to 2 years', '1969-77'), (v) **Numeral**, can be alphabetic numbers ('twenty-five', 'two') or combinations of numbers and words ('1 million', '3k'), (vi) **Number with Quantifier** ('>', 'less than', 'about'), (vii) **Percentage** is written either as '%' or 'percent'), (viii) **Roman numeral** ('iii', 'V').

Table 1 presents a summary of the prevalent types of numerical data, as well as examples of where each can be found in clinical texts.

3 Corpus analysis

Clinical documents are rich with diverse numerical data, expressed in different manners and contexts.

[1] This example is from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4311883/

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 409–415
August 16, 2024. ©2024 Association for Computational Linguistics

Type	Description	Instantiations in clinical texts
Cardinal	Used for counting or quantifying items in a set; total number of elements (Mirza et al., 2017).	number of participants sample size
Ordinal	Indicate the relative position or rank of an element within a series.	tumor stage
Measurement	A numerical value, typically accompanied by a unit, representing an attribute of a measured entity (Göpfert et al., 2022; Harper et al., 2021).	vital signs: body temperature, blood pressure, heart rate laboratory values: white blood cell count, hormone level, cholesterol
Temporal	Dates ('17 June 2024', '05/08/10'), times ('9pm', "two years ago"), and duration ("in an hour") (Tourille et al., 2017).	duration of intervention gestational age date ranges
Frequency	The number of times something occurs within a given interval.	medication dosage frequency
Proportion	A scaled quantity based on relative size	hospital readmission rate % group experiencing an outcome
Ratio	a comparison between two quantities	
Math	Numbers, variables, and operators in a mathematical statement such as a formula or probability; arithmetic operations as well as functions (Lu et al., 2023).	estimate of effect with confidence interval; p value
Non-numerical	Numerical values lacking number properties, e.g. as part of categorical data (identifiers) or or a named entities (e.g., COVID-19)	medical classification: disease code, pharmaceutical code

Table 1: Types of numerical information, and instantiations of each type in clinical text

To illustrate this, we empirically analyze four clinical NLP corpora. Our selection of corpora covers (i) various clinical NLP *tasks* — information extraction, information retrieval, natural language inference, and question answering — and (ii) *document types* — paper abstracts, clinical notes, and patient description narratives.

For each corpus, we present descriptive analysis. We count the frequency of numbers, estimated by how many digits and numerals occur in the text based on regular expression matching. We find that numbers are highly prevalent in each corpus. To understand contexts of number use, we sample and evaluate a few instances from each corpus qualitatively.

EBM-NLP (Nye et al., 2018) This corpus contains 4,993 abstracts of medical articles describing clinical randomized controlled trials in which text spans are annotated with PICO elements. That is, annotation labels include the trial (P)articipants enrolled, the (I)nterventions studied and to what they were (C)ompared, and the (O)utcomes measured. We find that 4,507 abstracts (90%) contain numerical information. The distribution of number token frequency with respect to number of abstracts is shown in Figure 1. The majority of abstract documents encode 5–20 numbers per abstract. However, only 13% of number tokens in the document collection are within annotated PICO-spans. About two-third of the numbers within spans belong to Participants entities, most of which relate to sample size and age of study population.

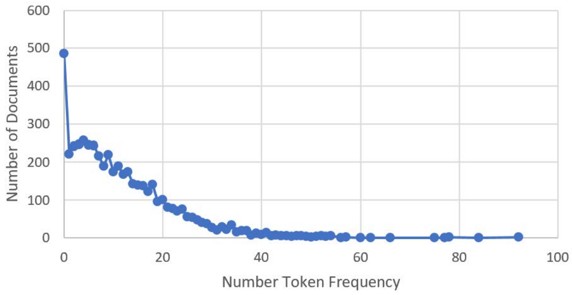


Figure 1: y EBM-NLP documents contain x numbers.

TREC-CDS (Koopman and Zuccon, 2016; Roberts et al., 2022) The TREC Clinical Trial series task involves matching a given patient to relevant clinical trials. The task is framed as retrieval of clinical trial documents using a patient descriptions as a topic query. The data includes 60 topics from

EBM-NLP (Nye et al., 2018)

METHODS We obtained economic data from 1424 Guatemalan individuals (aged 25–42 years) between 2002 and 2004. They accounted for 60% of the 2392 children (aged 0-7 years) . . . enrolled in a nutrition intervention study during 1969–77. In this initial study, two villages were randomly assigned a nutritious supplement (atole) for all children and two villages a less nutritious one (fresco). . . .

FINDINGS Exposure to atole before, but not after, age 3 years was associated with higher hourly wages, but only for men. For exposure to atole from 0 to 2 years, the increase was US$0.67 per hour (95% CI 0.16–1.17), which meant a 46% increase in average wages. There was a non-significant tendency for hours worked to be reduced and for annual incomes to be greater for those exposed to atole from 0 to 2 years.

TREC-CDS (Roberts et al., 2022)

Patient is a 55yo woman with h/o ESRD on HD and peritoneal dialysis who presented with watery, non bloody diarrhea and weakness. She has a history of 2 prior C diff infections, the most recent just 1 month ago. Recent antibx use in the last month on prior admission. Was also txd for Cdiff at that time for 14 d. course with po vanco. Pt was initially admitted to the ICU and was septic on pressors (levophed) until the morning of [**8–26**] with leukocytosis but no fever.

MedNLI (Romanov and Shivade, 2018)

Premise	The patient's hematocrit dropped from 29.7 to 22.8.
Hypothesis	The patient has a bleed.
Label	Entailment

Table 2: Sample instances involving numbers from EBM-NLP, TREC CDS, and MedNLI

TREC-CDS 2014 and 2015 (Koopman and Zuccon, 2016), 75 from TREC-CDS 2021, and 50 from 2022 (Roberts et al., 2022). We find that 100% of the patient descriptor topics contain numerical information. All of them contain patient age information expressed through different lexical variants. Most topics also contain numerical information about patient's vital signs, lab results, and medication history. Several types of numerical information are expressed as relations among measurement attributes, temporal information, and frequency.

MedNLI (Romanov and Shivade, 2018) Natural Language Inference (NLI) is an NLP task for determining whether a premise sentence semantically entails a hypothesis sentence. MedNLI is an NLI dataset sourced from clinical notes and annotated by a doctor. Each premise in MedNLI is grounded in the medical history of a patient and the hypothesis is a clinical conclusion labelled true, false, or maybe. We observe that nearly 50% of premise sentences contain numerical information, while only 1% of hypotheses have number tokens. This pattern is consistent across train, dev, and test data. We discover that numerical reasoning is one essential skill for formulating and interpreting medical conclusions.

PubMedQA (Jin et al., 2019) consists of a context and a yes/no/maybe question related to the context. Contexts are derived from PubMed abstracts, and questions are biomedical research questions. We find that 96.5% of contexts in the manually annotated subset PubMedQA-L contain numbers, including statistical information relating to trial results. Quantitative reasoning is needed to correctly infer the answer from the context.

4 Numeracy task and data in clinical domain

Thawani et al. (2021b) and Yoshida and Kita (2021) reviewed a broad range of numeracy tasks. Neither survey specifically considers the clinical domain. Given the findings of our corpus analysis (§3) that numbers are ubiquitous, we speculated that there may have been a number of related works on mining numerical information from clinical corpora. We search papers from the ACL Anthology[2] and PubMed[3] using the following keyword query.

> ("number" OR "numerical" OR "numeracy") AND ("clinical" OR "medical")

Among the numeracy tasks explored in the retrieved literature, mostly pertaining to information extraction, are extraction of lab test results (Bhatia et al., 2010; Liu et al., 2017) and extraction of measurement values from radiology report narrative (Bozkurt et al., 2019). In addition, we are aware that a number of clinical information extraction tasks also extract numerical attributes together with other entities, for example clinical trial variables (number of participants, sample size, outcome measurements) (Kiritchenko et al., 2010; Summerscales et al., 2011), eligibility criteria from clinical trials (Kury et al., 2020; Tseo et al., 2020),

[2] https://aclanthology.org/

[3] https://pubmed.ncbi.nlm.nih.gov/

medication attributes, such as drug dosage and frequency (Uzuner et al., 2010; MacKinlay and Verspoor, 2013; Kartchner et al., 2023), and temporal information (Sun et al., 2013; Styler IV et al., 2014; Miller et al., 2015)

Only a small number of tasks attempt to address numerical reasoning problems. One is NLI4CT, a dataset introduced for natural language inference and evidence retrieval tasks on clinical trial report (Jullien et al., 2023). Another addresses inference of patient phenotype based on extracted numerical values utilising one or more clinical attributes, e.g., "temperature 102°F suggesting Fever" (Tanwar et al., 2022).

5 Discussion

There are several possible directions to progress treatment of numbers in clinical NLP research.

Need for Benchmarks While there has been some research on extraction of numerical information from different medical data, most work has used their own data and the gold standard has not always been made public. This limitation has made it impossible to compare model performances of different systems (Jonnalagadda et al., 2015). The performance reported in several past works was very high (accuracy $>$ 90%). This raises the question of whether numerical information extraction is a solved task.

We raise two concerns. First, a number of works utilized relatively small data and this may result in the reported accuracy scores lacking statistical significance. Second, some works applied 'easy' task formulations, i.e., given a sentence containing only a single number mention, it is trivial to extract most numerical attributes. Such spurious patterns in evaluation data may not generalize when we deal with more realistic scenarios for information extraction (Elangovan et al., 2024). For example, a sentence containing multiple numbers and more than one candidate for entities and attributes (see example of EBM-NLP instance in Table 2). Hence, we advocate for more public data benchmarks to transparently evaluate the progress of numerical information tasks in the clinical domain.

Scope of Numerical Reasoning Recent works on numerical reasoning deal with math and arithmetic problems (Mishra et al., 2022; Hendrycks et al., 2021; Cobbe et al., 2021). In fact, complex mathematics are less applicable in the clinical domain. In addition to arithmetic, other types of reasoning are required for clinical decision support. For example, number comparison (Park et al., 2022) and number normalization (Almasian et al., 2023). Different units of measurement often need conversion, requiring precise calculations to maintain accuracy. For example, hormone levels may be measured using nmol/L, ng/dL and ng/mL in different trials. Dealing with various number representation is important to interpret numerical information correctly. On the other hand, contextualizing such numbers with medical background knowledge is another important numeracy skill, as showcased by the example of MedNLI instance in Table 2.

Tokenization Challenges How to encode numbers in language models has been discussed in several works (Spithourakis and Riedel, 2018; Wallace et al., 2019; Thawani et al., 2021a). Encoding numbers is related to the problem of tokenization (Geva et al., 2020). The models struggle to recognize extrapolated numbers that are seldom found in corpora and force them to be tokenized digit by digit (Kim et al., 2021). In the clinical context, when the numbers are grounded with units or appear as ranges, the tokenizer is expected to be more robust. We inspected few samples of token-level annotated data of the EBM-NLP corpus. Our finding was that numbers are not fully correctly tokenized (e.g., "95% CI 0.16-1.17" is segmented into multiple individual digit numbers that lack meaning), even in the gold standard.

Utilizing Numerical Information for Clinical Application. Clinical documents contain numerous numerical data points. However, numbers are mostly neglected when designing an NLP system (Thawani et al., 2021b). Entity annotations skip numbers in most cases, as in the EBM-NLP corpus PICO annotation (Nye et al., 2018). Several clinical NLP works acknowledge the importance of numerical reasoning, but leave it for future work. For instance, in multi-document summarization, Otmakhova et al. (2022) identify that automatically generating systematic reviews involves meta-analysis that requires numerical aggregation of data across primary studies or calculating some statistics for variables. In another example, Lehman et al. (2019) argue that numerical information from the result section of studies can be utilized to improve evidence inference.

6 Conclusions

We analyzed well-established clinical NLP corpora, covering a variety of tasks and data sources, and identifying a broad set of types and usage of numbers. Our analysis shows that numbers play a major role in medical texts. On the basis of these findings, and the lack of systematic resources in the clinical domain for investigating numerical information extraction and reasoning tasks, we argue for the need for the construction of such resources. Numbers contain vital medical information. We strongly encourage clinical NLP researchers to consider how numerical processing may interact with their work.

Limitation

Our conclusion is based on the corpus we analyzed and reviewed during literature search. We may not include some corpora, especially those that are not publicly available, in our analysis. On the other hand, this work focuses only on English. While there have been some relevant works in the clinical domain for languages other than English, we leave this for future work.

Acknowledgments

The authors would like to acknowledge Country. This research has been carried out on the unceded lands of the Woi Wurrung and Boon Wurrung language groups of the eastern Kulin nation. We pay our respects to their Ancestors and Elders, past, present, and emerging. We respectfully acknowledge First Nations' connection to land, waters, sky, and community. This research was funded by the Australian Research Council through an Industrial Transformation Training Center Grant (grant IC170100030).

References

Satya Almasian, Vivian Kazakova, Philipp Göldner, and Michael Gertz. 2023. CQE: A comprehensive quantity extractor. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12845–12859, Singapore. Association for Computational Linguistics.

Ramanjot Singh Bhatia, Amber Graystone, Ross A Davies, Susan McClinton, Jason Morin, and Richard F Davies. 2010. Extracting information for generating a diabetes report card from free text in physicians notes. In *Proceedings of the NAACL HLT 2010 Second Louhi Workshop on Text and Data Mining of Health Documents*, pages 8–14, Los Angeles, California, USA. Association for Computational Linguistics.

Selen Bozkurt, Emel Alkim, Imon Banerjee, and Daniel L. Rubin. 2019. Automated detection of measurements and their descriptors in radiology reports using a hybrid natural language processing algorithm. *Journal of Digital Imaging*, 32(4):544–553.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Aparna Elangovan, Jiayuan He, Yuan Li, and Karin Verspoor. 2024. Principles from clinical research for NLP model generalization. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics*, page to appear, Mexico City, Mexico. Association for Computational Linguistics.

Mor Geva, Ankit Gupta, and Jonathan Berant. 2020. Injecting numerical reasoning skills into language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 946–958, Online. Association for Computational Linguistics.

Jan Göpfert, Patrick Kuckertz, Jann Weinand, Leander Kotzur, and Detlef Stolten. 2022. Measurement extraction with natural language processing: A review. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2191–2215, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

David A. Hanauer, Qiaozhu Mei, V. G. Vinod Vydiswaran, Karandeep Singh, Zach Landis-Lewis, and Chunhua Weng. 2019. Complexities, variations, and errors of numbering within clinical notes: the potential impact on information extraction and cohort-identification. *BMC Medical Informatics and Decision Making*, 19(3):75.

Corey Harper, Jessica Cox, Curt Kohler, Antony Scerri, Ron Daniel Jr., and Paul Groth. 2021. SemEval-2021 task 8: MeasEval – extracting counts and measurements and their related contexts. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 306–316, Online. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *NeurIPS*.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the*

9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

Siddhartha R. Jonnalagadda, Pawan Goyal, and Mark D. Huffman. 2015. Automating data extraction in systematic reviews: a systematic review. *Systematic Reviews*, 4(1):78.

Mael Jullien, Marco Valentino, Hannah Frost, Paul O'Regan, Dónal Landers, and Andre Freitas. 2023. NLI4CT: Multi-evidence natural language inference for clinical trial reports. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16745–16764, Singapore. Association for Computational Linguistics.

David Kartchner, Selvi Ramalingam, Irfan Al-Hussaini, Olivia Kronick, and Cassie Mitchell. 2023. Zero-shot information extraction for clinical meta-analysis using large language models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 396–405, Toronto, Canada. Association for Computational Linguistics.

Jeonghwan Kim, Giwon Hong, Kyung-min Kim, Junmo Kang, and Sung-Hyon Myaeng. 2021. Have you seen that number? investigating extrapolation in question answering models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7031–7037, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Svetlana Kiritchenko, Berry de Bruijn, Simona Carini, Joel Martin, and Ida Sim. 2010. ExaCT: automatic extraction of clinical trial characteristics from journal publications. *BMC Med. Inform. Decis. Mak.*, 10(1):56.

Bevan Koopman and Guido Zuccon. 2016. A test collection for matching patients to clinical trials. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '16, page 669–672, New York, NY, USA. Association for Computing Machinery.

Fabrício Kury, Alex Butler, Chi Yuan, Li-heng Fu, Yingcheng Sun, Hao Liu, Ida Sim, Simona Carini, and Chunhua Weng. 2020. Chia, a large annotated corpus of clinical trial eligibility criteria. *Scientific Data*, 7(1):281.

Eric Lehman, Jay DeYoung, Regina Barzilay, and Byron C. Wallace. 2019. Inferring which medical treatments work from reports of clinical trials. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3705–3717, Minneapolis, Minnesota. Association for Computational Linguistics.

Sijia Liu, Liwei Wang, Donna M. Ihrke, Vipin Chaudhary, Cui Tao, Chunhua Weng, and Hongfang Liu. 2017. Correlating lab test results in clinical notes with structured lab data: A case study in hba1c and glucose. *AMIA Summits on Translational Science Proceedings*, pages 221 – 228.

Pan Lu, Liang Qiu, Wenhao Yu, Sean Welleck, and Kai-Wei Chang. 2023. A survey of deep learning for mathematical reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14605–14631, Toronto, Canada. Association for Computational Linguistics.

Andrew MacKinlay and Karin Verspoor. 2013. Information extraction from medication prescriptions within drug administration data. In *The 4th International Workshop on Health Document Text Mining and Information Analysis with the focus of Cross-language Evaluation (LOUHI), Canberra/Sydney, Australia*.

Timothy Miller, Steven Bethard, Dmitriy Dligach, Chen Lin, and Guergana Savova. 2015. Extracting time expressions from clinical text. In *Proceedings of BioNLP 15*, pages 81–91, Beijing, China. Association for Computational Linguistics.

Kristian Miok, Padraig Corcoran, and Irena Spasić. 2023. The value of numbers in clinical text classification. *Machine Learning and Knowledge Extraction*, 5(3):746–762.

Paramita Mirza, Simon Razniewski, Fariz Darari, and Gerhard Weikum. 2017. Cardinal virtues: Extracting relation cardinalities from text. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 347–351, Vancouver, Canada. Association for Computational Linguistics.

Swaroop Mishra, Matthew Finlayson, Pan Lu, Leonard Tang, Sean Welleck, Chitta Baral, Tanmay Rajpurohit, Oyvind Tafjord, Ashish Sabharwal, Peter Clark, and Ashwin Kalyan. 2022. LILA: A unified benchmark for mathematical reasoning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5807–5832, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Benjamin Nye, Junyi Jessy Li, Roma Patel, Yinfei Yang, Iain Marshall, Ani Nenkova, and Byron Wallace. 2018. A corpus with multi-level annotations of patients, interventions and outcomes to support language processing for medical literature. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 197–207, Melbourne, Australia. Association for Computational Linguistics.

Yulia Otmakhova, Karin Verspoor, Timothy Baldwin, Antonio Jimeno Yepes, and Jey Han Lau. 2022. M3: Multi-level dataset for multi-document summarisation of medical studies. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3887–3901, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Sungjin Park, Seungwoo Ryu, and Edward Choi. 2022. Do language models understand measurements? In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1782–1792, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kirk Roberts, Dina Demner-Fushman, Ellen M. Voorhees, Steven Bedrick, and William R. Hersh. 2022. Overview of the TREC 2022 clinical trials track. In *Proceedings of the Thirty-First Text REtrieval Conference, TREC 2022, online, November 15-19, 2022*, volume 500-338 of *NIST Special Publication*. National Institute of Standards and Technology (NIST).

Alexey Romanov and Chaitanya Shivade. 2018. Lessons from natural language inference in the clinical domain. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1586–1596, Brussels, Belgium. Association for Computational Linguistics.

Georgios Spithourakis and Sebastian Riedel. 2018. Numeracy for language models: Evaluating and improving their ability to predict numbers. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2104–2115, Melbourne, Australia. Association for Computational Linguistics.

William F. Styler IV, Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet C de Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, and James Pustejovsky. 2014. Temporal annotation in the clinical domain. *Transactions of the Association for Computational Linguistics*, 2:143–154.

Rodney L. Summerscales, Shlomo Argamon, Shangda Bai, Jordan Hupert, and Alan Schwartz. 2011. Automatic summarization of results from clinical trials. In *2011 IEEE International Conference on Bioinformatics and Biomedicine*, pages 372–377.

Weiyi Sun, Anna Rumshisky, and Ozlem Uzuner. 2013. Evaluating temporal relations in clinical text: 2012 i2b2 challenge. *J. Am. Med. Inform. Assoc.*, 20(5):806–813.

Ashwani Tanwar, Jingqing Zhang, Julia Ive, Vibhor Gupta, and Yike Guo. 2022. Phenotyping in clinical text with unsupervised numerical reasoning for patient stratification. *Exp. Biol. Med. (Maywood)*, 247(22):2038–2052.

Avijit Thawani, Jay Pujara, and Filip Ilievski. 2021a. Numeracy enhances the literacy of language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6960–6967, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Avijit Thawani, Jay Pujara, Filip Ilievski, and Pedro Szekely. 2021b. Representing numbers in NLP: a survey and a vision. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–656, Online. Association for Computational Linguistics.

Julien Tourille, Olivier Ferret, Xavier Tannier, and Aurélie Névéol. 2017. Temporal information extraction from clinical text. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 739–745, Valencia, Spain. Association for Computational Linguistics.

Yitong Tseo, M. I. Salkola, Ahmed Mohamed, Anuj Kumar, and Freddy Abnousi. 2020. Information extraction of clinical trial eligibility criteria. *CoRR*, abs/2006.07296.

Ozlem Uzuner, Imre Solti, and Eithon Cadag. 2010. Extracting medication information from clinical text. *J. Am. Med. Inform. Assoc.*, 17(5):514–518.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. 2019. Do NLP models know numbers? probing numeracy in embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5307–5315, Hong Kong, China. Association for Computational Linguistics.

Minoru Yoshida and Kenji Kita. 2021. Mining numbers in text: A survey. *Information Systems-Intelligent Information Processing Systems, Natural Language Processing, Affective Computing and Artificial Intelligence, and an Attempt to Build a Conversational Nursing Robot*.

A Fine-grained citation graph for biomedical academic papers: the finding-citation graph

Yuan Liang
Queen Mary University
London, UK
yuan.liang@qmul.ac.uk

Roonak Rezvani
Exscientia
Oxford, UK
rrezvani@exscientia.co.uk

Massimo Poesio
Queen Mary University
London, UK
m.poesio@qmul.ac.uk

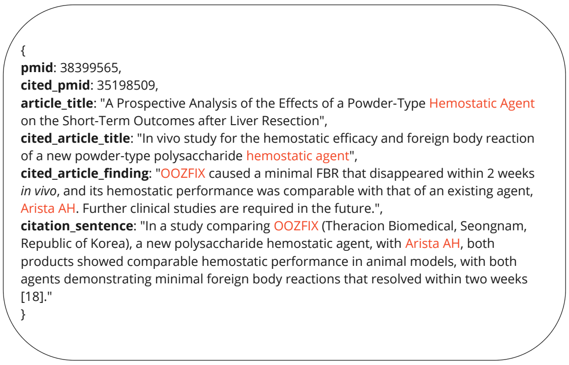
Abstract

Citations typically mention findings as well as papers. To model this richer notion of citation, we introduce a richer form of citation graph with nodes for both academic papers and their findings: the finding-citation graph (FCG). We also present a new pipeline to construct such a graph, which includes a finding identification module and a citation sentence extraction module. From each paper, it extracts rich basic information, abstract, and structured full text first. The abstract and vital sections, such as the results and discussion, are input into the finding identification module. This module identifies multiple findings from a paper, achieving an 80% accuracy in multiple findings evaluation. The full text is input into the citation sentence extraction module to identify inline citation sentences and citation markers, achieving 97.7% accuracy. Then, the graph is constructed using the outputs from the two modules mentioned above. We used the Europe PMC to build such a graph using the pipeline, resulting in a graph with 14.25 million nodes and 76 million edges.

1 Introduction

In recent years, the volume of biomedical literature has been constantly growing. More than 3000 articles are published every day on average and PubMed alone has a total of 29M articles as of January 2019 (Lee et al., 2019). This makes it difficult for experts to understand and assess the publications within a short amount of time.

Citations play a crucial role in academic papers, linking the new work to related research (Cohan et al., 2019). They can assist in evaluating research outputs (Yue and Wilson, 2004), and tracking the progression of research while predicting future directions (Prabhakaran et al., 2016). The citation network, a graph that records the citation relationship between papers, is commonly used in such studies (Gundolf and Filser, 2013; Hota et al., 2020; Zhao, 2020) .



{
pmid: 38399565,
cited_pmid: 35198509,
article_title: "A Prospective Analysis of the Effects of a Powder-Type Hemostatic Agent on the Short-Term Outcomes after Liver Resection",
cited_article_title: "In vivo study for the hemostatic efficacy and foreign body reaction of a new powder-type polysaccharide hemostatic agent",
cited_article_finding: "OOZFIX caused a minimal FBR that disappeared within 2 weeks *in vivo*, and its hemostatic performance was comparable with that of an existing agent, Arista AH. Further clinical studies are required in the future.",
citation_sentence: "In a study comparing OOZFIX (Theracion Biomedical, Seongnam, Republic of Korea), a new polysaccharide hemostatic agent, with Arista AH, both products showed comparable hemostatic performance in animal models, with both agents demonstrating minimal foreign body reactions that resolved within two weeks [18]."
}

Figure 1: An example of the relation between paper and cited paper's finding through the citation sentence.

In recent years, many academic databases, which also can be regarded as academic citation networks, have been developed to facilitate detailed citation studies on biomedical publications. They provide basic information for hundreds of millions of academic papers and the citations between these documents. Some of these databases are commercial, like Clarivate's Web of Science (WoS) and Elsevier's Scopus. Others are open-source, in line with current trends. These include Microsoft Academic Graph (MAG) (Sinha et al., 2015), OpenCitations Index of CrossRef open DOI-DOI citations (COCI) (Heibi et al., 2019), Dimensions (Herzog et al., 2020), National Institutes of Health's Open Citation Collection (NIH-OCC) (Ian Hutchins et al., 2019), Semantic Scholar's Open Research Corpus (S2ORC) (Lo et al. 2020; Kinney et al. 2023). Some statistics about these databases are shown in Table 1.

From Table 1, it is clear that all databases, except S2ORC, lack inline citation contexts. These contexts provide information on what and why a paper cites information from other papers. For example, it may cite another paper's findings or refer to background or statistical information (Cohan et al., 2019). The findings of the paper are the most valuable output of the study. Only cita-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 416–426
August 16, 2024. ©2024 Association for Computational Linguistics

Database	Version	Publication	Citation	Access	Disciplines	Citation Contexts
Wos Core	2024	92M	2.2BN	Commercial	Multi	No
Scopus	2024	94M	2.4B	Commercial	Multi	No
MAG	2020-10	240M	-	Stop Serving	Multi	No
COCI	2023-01	77M	1.4B	Open Source	Multi	No
Dimensions	2024	140M	-	Application Needed	Multi	No
NIH-OCC	2024-04	37M	782M	Open Source	Health	No
S2ORC	2024	214M	2.49B	Application Needed	Multi	Yes

Table 1: A comparison between existing academic databases covering medical corpus.

tions of these findings can be used to evaluate the value of the publication and the research output. However, to understand whether a paper cites another paper's findings and which specific finding it refers to, the research findings need to be identified. A relationship between the citations and findings must also be established. An example can be seen in the Figure 1. Existing research on identifying research findings, such as the approach proposed by Wright et al. (2022) to extract sentences describing research findings and study their dissemination in scientific communication, can be helpful in this context. Motivated by the challenges of current databases and the existing research on finding identification, we propose the development of a fine-grained citation graph. This graph will involve both research findings and citation contexts. It will enable detailed evaluation and study of finding evolution from the citation perspective.

In this paper, we define the finding-citation graph first in section 3. Then, we outline the process of constructing the finding-citation graph using the European PMC dataset in section 4. We also evaluate the construction to ensure quality. The summary statistics of the graph and some interesting observations are presented in section 5.

2 Related Work

Constructing a fine-grained citation graph directly relates to cite-worthiness detection, and finding identification. We will briefly introduce these aspects in the following sub-sections. Since the biomedical large language model (LLM) has recently gained popularity and may be used in our project, it will also be introduced in the subsequent sub-sections.

Cite-Worthiness Detection Cite-worthiness detection involves identifying citation sentences in an academic paper. These sentences contain references to external sources cited within the paper. There are many different forms of citation, but the most common are:

- The topic is studied in previous work (Author1 et al. ###).
- The topic is studied in previous work [##].
- The topic is studied in previous work (##).
- The topic is studied using XXX (Author 1 et al. ###) and XXX (Author1 et al. ###) XXX.
- Author 1 et al, ### (year) performs XXX.

Sugiyama, Kumar, Kan, and Tripathi (2010) suggested the application of Support Vector Machines (SVMs) with diverse features for cite-worthiness detection. These features range from unigrams, bigrams, and the existence of proper nouns, to section information, classification of neighboring sentences, and orthographic checking. They designed a dataset using the ACL Anthology Reference Corpus (ACL-ARC) (Bird et al., 2008), applying regular expression patterns. Similarly, Färber et al. (2018b) carried out the same task using convolutional recurrent neural networks on an expanded dataset. The dataset incorporated three subsets: ACL-ARC (Bird et al., 2008), arXiv CS (Färber et al., 2018a), and Scholarly Dataset 2.

However, the datasets from these studies are confined to one or a limited number of domains and exhibit a high class imbalance. As per Färber et al. (2018b), only a tenth of all sentences hold at least one citation marker, leaving the remaining 90% without any. Furthermore, these studies lack an in-depth discussion on dataset creation and qualitative analysis.

In response to these issues, Wright and Augenstein (2021) introduced a dataset for spotting citation-worthy sources across six domains. They detailed the process for creating the dataset and provided a qualitative analysis. However, their approach to dataset creation was limited to using regular expressions to identify the first and second citation forms mentioned above. Besides, the authors trained a set of baseline models on their dataset to evaluate performance and understand the complexity of the problem. The results of these models are displayed in Fig 2.

Method	P	R	F1
Logistic Regression	$46.65_{0.00}$	$64.88_{0.00}$	$54.28_{0.00}$
Färber et al. (2018b)	$49.57_{0.96}$	$65.56_{2.61}$	$56.41_{0.34}$
Transformer	$47.92_{0.78}$	$71.59_{1.74}$	$57.39_{0.10}$
BERT	$55.04_{0.66}$	$69.02_{1.33}$	$61.23_{0.21}$
SciBERT-no-weight	$\mathbf{65.94}_{0.37}$	$51.62_{0.53}$	$57.91_{0.30}$
SciBERT	$57.03_{0.50}$	$68.08_{1.03}$	$62.06_{0.15}$
SciBERT + PU	$49.46_{0.83}$	$\mathbf{82.12}_{1.40}$	$61.73_{0.27}$
Longformer-Solo	$57.21_{0.25}$	$68.00_{0.41}$	$62.14_{0.02}$
Longformer-Ctx	$59.92_{0.28}$	$77.15_{0.49}$	$\mathbf{67.45}_{0.06}$

Figure 2: Performance of models on the CITEWORTH dataset (Wright and Augenstein, 2021)

Finding Identification The process of pinpointing and extracting results or conclusions from an academic paper is known as finding identification. Prabhakaran, Hamilton, McFarland, and Jurafsky (2016) designed a Conditional Random Field (CRF) model that manages sentence-level sequence labeling, designating each sentence in the abstract a rhetorical role, including result and conclusion. Dernoncourt and Lee (2017) introduced a considerable sentence classification dataset, PubMed 200K RCT. This dataset, consisting of roughly 200,000 abstracts of randomized controlled trials (RCTs) and a total of 2.3 million sentences, labels each sentence with its rhetorical role, which includes the result and conclusion. Though it is limited to the RCT field, this dataset can help find identification. Inspired by the PubMed 200K RCT dataset, Wright et al. (2022) curated a dataset of 200K self-labeled abstracts from PubMed, with no field restrictions. Then, they fine-tuned a RoBERTa model (Liu et al., 2019) on this dataset, classifying each sentence in the abstract into categories such as result, conclusion, method, background, and others. The model achieved an F1 score of 92%, and when applied to the full text of papers, it performed well. Previous studies have overlooked the importance of certain sections of papers, such as the results and conclusion sections. These sections often contain important findings. Past studies mainly focus on finding extraction, neglecting the potential for finding generation. However, the development of large generative language models, like Llama (Touvron et al., 2023), now provides the opportunity for effective finding generation.

In the finding identification task, there is a subtask known as claim identification or argumentation mining exists. According to the definition from Achakulvisut et al. (2020b), a claim is (1) a statement declaring something as superior, (2) a statement proposing something new, or (3) a statement describing a new discovery or a new cause-effect relationship. The definition of a claim differs from that of a finding, being stricter and more precise. Nonetheless, some ideas from this research could be useful. Achakulvisut et al. (2020b) developed a tool for annotating claims and collected 1500 labeled abstracts (SciCE) from PubMed articles published from 2008 to 2018. These abstracts incorporate 11,702 sentences in total, with each sentence labeled as a claim or non-claim. This tool effectively tackles the issue of data scarcity in the task. They also constructed a new model incorporating transfer learning, which improved the F1 score by 14 percentage points compared to the baseline model without transfer learning. In 2023, Wei et al. undertook the same task, achieving a new state-of-the-art (SOTA) performance on the dataset using supervised contrastive learning and transfer learning, with an 87.45% F1 score. As observed, all these models operate on the abstract rather than the full-text article, and the shift to the full-text article still poses a challenge due to the writing structure of the complete publication.

Biomedical LLMs Back in 2018, ELMo (Peters et al., 2018) pioneered the use of a context-sensitive language model pre-trained on a huge data corpus. This sparked a wave of LLMs such as GPT (Radford and Narasimhan, 2018), BERT (Devlin et al., 2019), ERNIE (Zhang et al., 2019), GPT-2 (Radford et al., 2019), GPT3 (Brown et al., 2020), among others. These LLMs are incredibly useful for a variety of natural language processing (NLP) tasks. However, general-purpose LLMs, which are trained on resources like English Wikipedia and BookCorpus, often struggle with biomedical NLP tasks due to the numerous domain-specific terms and proper nouns. To counter this, many LLMs have been pre-trained on biomedical corpus like PubMed abstracts and PubMed Central full-text articles, enhancing their performance in the biomedical field.

There are two main approaches to domain-specific neural language model paradigms: mix-domain pre-training and domain-specific pre-training from scratch. Mix-domain pre-training, such as BioBERT (Lee et al., 2019) and BlueBERT (Peng et al., 2019), begins with parameters from a general-purpose language model and adopts its vocabulary. On the other hand, domain-specific pre-training from scratch, like PubMedBERT (Gu et al., 2021), BioLinkBERT (Yasunaga et al., 2022), BioMedLM (Bolton et al., 2024), and Bioformer-8L (Fang et al., 2023), generates vocabulary and conducts pre-training using only the in-domain corpus. Models like PubMedBERT and BioMedLM have shown that domain-specific pre-training from scratch can outperform mix-domain pre-training.

3 The Finding-Citation Graph: Definition

Building on the work of Wright et al. (2022), we define a finding as a statement that describes a specific research outcome from a scientific study. We also describe a citation sentence as a sentence that references knowledge from other papers.

Subsequently, we define a finding-citation graph (FCG) as $G = (P, F, C, B)$, where P, F, C, and B represent sets of papers, findings, citations, and basic information respectively. A paper in this graph is an academic paper. A finding in this graph is a statement same as the above definition. A citation within the graph refers to instances where the citation sentence includes the findings of the cited paper, which we will now refer to as a useful citation. The basic information includes the author, journal, publication year, etc. of the paper containing the finding. The defined finding-citation graph is a heterogeneous graph and can be perceived as a variant of the citation graph, where the node is the paper and the relation is the citation.

4 Constructing the FCG

We now introduce our pipeline to construct the finding-citation graph (Figure 3), which allows us to analyze findings from the perspective of the citation network. The pipeline takes a Europe PubMed article in XML format as input and produces three types of information for each paper: basic information, all citation sentences, and all findings. This information is then used to construct the graph. The pipeline comprises four main modules, as follows:

- An XML parser is utilized to extract essential paper information and article content from the XML. The primary components of the article include the abstract and the full-text article, composed of various sections.
- The finding identification model aims to identify sentences that describe findings from the abstract, conclusion, and result sections.
- The citation sentence extraction module identifies sentences within the full-text article that contain citations and links the citation sentences with its cited paper.
- The final module is to build the finding-citation graph construction based on the output of the above three modules

The above four modules will be introduced in the following sections excluding the XML parser. Our parser was primarily based on an open-source PubMed parser (Achakulvisut et al., 2020a), with minor changes made to increase speed.

4.1 Finding Identification

This module includes two steps:

- Identify the sentences that discuss the findings, which are called finding sentences later.
- Generate findings based on the identified finding sentences, which are called findings later.

We performed the first step similar to Prabhakaran et al. (2016) and Wright et al. (2022), where the task was treated as a sentence classification task. They classified sentences in the abstract into five classes: background, result, conclusion, method, and objective. Sentences labeled as the result or conclusion can be considered findings sentences. To build the sentence classification model, we curated a dataset from self-annotated PubMed abstracts, as shown in Figure 4. After filtering for PubMed abstracts that met the set format, we obtained 206K suitable abstracts, comprising roughly 2.5 million labeled sentences. We then fine-tuned a RoBERTa model (Liu et al., 2019) on this curated dataset, achieving an accuracy score of 91% on a held-out 13.5% sample. This classifier was applied to the abstract sentences and other sections of the paper, like the results and conclusion, generating multiple finding sentences for each paper.

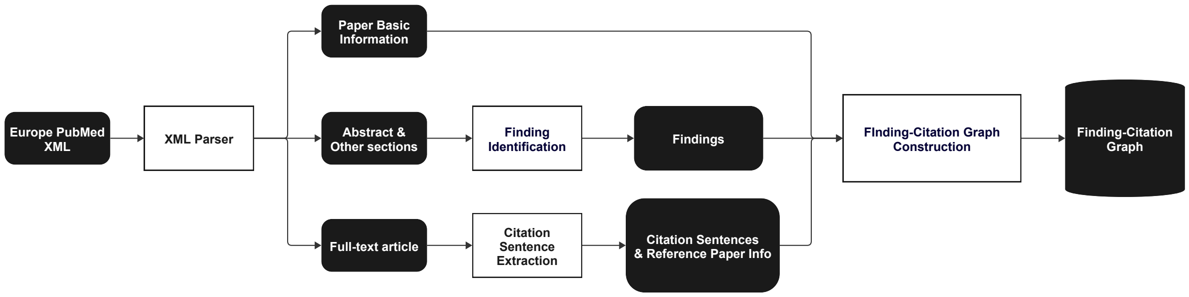
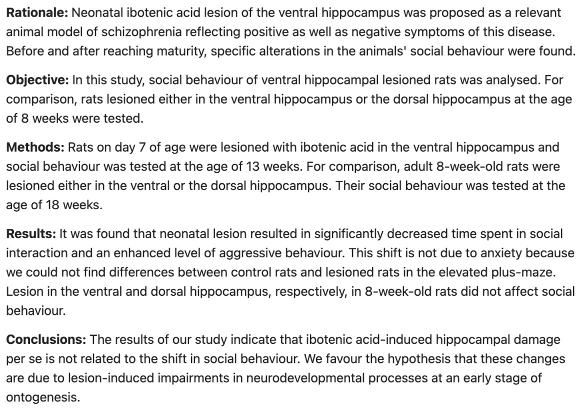

Paper Basic Information
Europe PubMed XML
XML Parser
Abstract & Other sections
Finding Identification
Findings
FInding-Citation Graph Construction
Finding-Citation Graph
Full-text article
Citation Sentence Extraction
Citation Sentences & Reference Paper Info

Figure 3: Finding-citation graph construction pipeline

We also experimented with LLMs to assess their potential as a substitute for the current fine-tuned model. Specifically, we utilized Gemma (Gemma Team et al., 2024) to assist us in categorizing the information in the abstract into the five classes mentioned above. To evaluate Gemma's performance, we compared the organized information for each class with self-annotated information, calculating similarity scores. When we set 0.5 as the similarity threshold, the accuracy was approximately 91%. As the performance is nearly the same, taking the time-consuming and resource-consuming into consideration, we chose to use fine-tuned RoBERTa model in our pipeline.



Rationale: Neonatal ibotenic acid lesion of the ventral hippocampus was proposed as a relevant animal model of schizophrenia reflecting positive as well as negative symptoms of this disease. Before and after reaching maturity, specific alterations in the animals' social behaviour were found.

Objective: In this study, social behaviour of ventral hippocampal lesioned rats was analysed. For comparison, rats lesioned either in the ventral hippocampus or the dorsal hippocampus at the age of 8 weeks were tested.

Methods: Rats on day 7 of age were lesioned with ibotenic acid in the ventral hippocampus and social behaviour was tested at the age of 13 weeks. For comparison, adult 8-week-old rats were lesioned either in the ventral or the dorsal hippocampus. Their social behaviour was tested at the age of 18 weeks.

Results: It was found that neonatal lesion resulted in significantly decreased time spent in social interaction and an enhanced level of aggressive behaviour. This shift is not due to anxiety because we could not find differences between control rats and lesioned rats in the elevated plus-maze. Lesion in the ventral and dorsal hippocampus, respectively, in 8-week-old rats did not affect social behaviour.

Conclusions: The results of our study indicate that ibotenic acid-induced hippocampal damage per se is not related to the shift in social behaviour. We favour the hypothesis that these changes are due to lesion-induced impairments in neurodevelopmental processes at an early stage of ontogenesis.

Figure 4: An example of a self-annotated PubMed abstract from PubMed PMID:10435405.

It is important to note that these finding sentences may contain overlap information with each other and may discuss multiple discoveries from the paper. Additionally, not all findings carry equal importance to the article. Our next step involved generating multiple findings for each paper, keeping these points in mind. We used a combination of scientific sentence-BERT (Wright et al., 2022) and the Affinity Propagation clustering method to eliminate duplicate sentences and select central sentences as representative findings. This approach yielded multiple findings (maximum 3) from each paper. Afterward, we computed the similarity score between each finding and the title of its corresponding article. This score is considered as the importance score for each finding within its respective paper. Consequently, we obtained multiple findings for each paper along with an importance ranking score. This procedure is illustrated in Figure 5. In order to know how the multi-finding module performs, we randomly sampled 100 articles with 241 findings and got 80% accuracy. We found that some errors originated from the abstract's conclusion sentence, which did not accurately represent the actual conclusions and simply offered a concluding sentence without any useful information.

4.2 Citation Sentence Extraction

The task involves identifying sentences in the article that reference external knowledge from other papers. Unlike other researchers such as Sugiyama et al. (2010) and Färber et al. (2018b), who employed binary classification models for this task, we used a simpler yet effective method: the regular expression. We addressed three formats of citation using this method, shown below. The use of regular expression simplified the process of linking the citation sentence with its cited paper. This was based on the citation marker and reference information derived from the XML. Consequently, we obtained citation sentences along with the information of the cited paper for each article.

- The topic is studied ... (Author1 et al. ###).
- The topic is studied ... [###].
- The topic is studied ... (###).

To evaluate how the module performed and maximize the use of the open-source dataset, we designed the following two-step evaluation method.

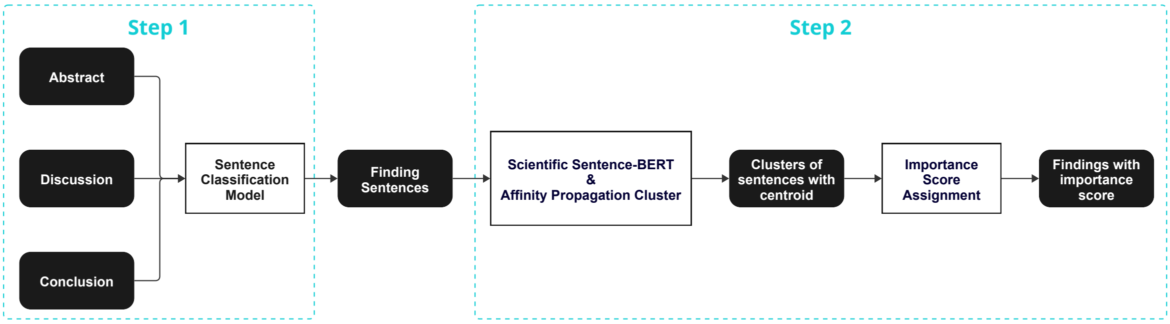

Step 1
Abstract
Discussion
Conclusion
Sentence Classification Model
Finding Sentences
Step 2
Scientific Sentence-BERT & Affinity Propagation Cluster
Clusters of sentences with centroid
Importance Score Assignment
Findings with importance score

Figure 5: Finding Identification Procedure

The first is a paper-level evaluation, which checks the accuracy of the citation relations, i.e., the PMID-PMID relation. Same as Liang et al. (2021), we utilized the PMC dataset in PubMed Baseline as the gold standard for this evaluation. Both the original and filtered PMID-PMID relations of the module were evaluated, with the latter excluding references not found in the Europe PMC dataset. In order to do the comparison with other open-source datasets, we used the same evaluation metrics as Liang et al. (2021), which are precision, recall, F1-score, and accuracy. It should be noted that only articles covered by the data source were included in the evaluation process. The evaluation results can be seen in Table 2. Even though our performance is not bad, our precision and recall are not the best among all the databases because the citation relations were based on citation sentences and markers, not the reference list. This discrepancy may lead to errors and losses in citation relations.

The second evaluation is to assess the correctness of the tuple, $(citing\_pmid, citation\_sentence, cited\_pmid)$. This formed the final output of the module. There is no other database containing the citation sentences on the PubMed corpus, except for S2ORC (Lo et al., 2020). However, the performance evaluation from Step 1 indicates that the S2ORC database did not perform well. Moreover, the S2ORC paper (Lo et al., 2020) does not provide a significant evaluation of the citation sentence, so we do not use it for evaluation. We randomly sampled 350 tuples and achieved 97.7% accuracy. We conducted an analysis to determine why certain tuples are incorrect. We found that some errors arise from mismatches between the description citation marker and the basic information of the cited paper. Other errors occur when the citation sentences are correctly identified, but the PMID of the cited paper is lost. This largely confirms our previous analysis above that the citations are based on citation sentences and citation markers can lead to errors and losses in this module.

4.3 Finding-Citation Graph Construction

So far, we have collected multiple findings for each paper, along with their importance scores and citation sentences with basic information about the cited paper. Using this information, we can create the finding-citation graph as outlined in Section 3.1. As we construct the graph based on a closed dataset, Europe PMC, the references without PMID or not in the closed dataset were dropped.

The graph comprises two types of nodes: articles and findings. Each article node has some basic attributes such as authors, journal, paper PMID, title, and publication year. In contrast, the finding node has no other attributes. The graph also contains two kinds of edges. The first represents the relationship between an article and its findings, with the importance score as an attribute. The second represents the citation relationship between an article and the findings produced by the cited paper, with the citation sentence and similarity score as attributes. We calculate the similarity score using a fine-tuned scientific sentence-BERT (Wright et al., 2022). This approach helps us determine whether a citation sentence contains the findings of another paper and assess the usefulness of each citation. A simplified view of the graph can be seen in Figure 6.

5 Experiments

We utilized Europe PMC articles in XML format for our experiment. Europe PMC is an open-source, global biomedical literature repository that houses life science articles, preprints, micropublications, books, patents, and clinical guidelines from around the world. Up to Feb 2024, it holds over 40 million

Metrics	COCI.Updated	Dimensions	NIH-OCC	S2ORC	S2ORC_new	Our_O	Our_D
Precision	98.82%	99.60%	99.9%	97.66%	77%	94.32%	92.35%
Recall	85.18%	98.80%	98.99%	79.00%	25.4%	89.65%	93.05%
F1-score	90.95%	99.07%	99.34%	86.27%	34.5%	89.32%	90.7%
Accuracy	15.60%	81.55%	89.08%	5.86%	1.03%	34.4%	55.37%

Table 2: The evaluation of COCI.Updated, Dimensions, NIH-OCC, and S2ORC are from Liang et al. (2021). Our main comparison is S2ORC, which is the most similar database to use and includes citation contexts, so we evaluated S2ORC on the latest version again. It is not only for the comparison but also for the confirmation of our comparison. The Our_O and Our_D are the original and filtered PMID-PMID relations respectively. When we did the evaluation on the filtered PMID-PMID relations, we did the same filter to the gold dataset.

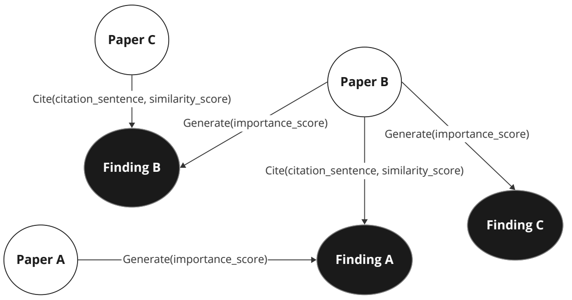

Paper C
Cite(citation_sentence, similarity_score)
Paper B
Generate(importance_score)
Generate(importance_score)
Finding B
Cite(citation_sentence, similarity_score)
Finding C
Paper A
Generate(importance_score)
Finding A

Figure 6: A simplified view of the finding-citation graph

Type of Article	Count(%)
`Europe PMC XML`	6M (100%)
`Successfully parsed`	5.75M (95.8%)
`With PMCID`	5.75M (95.8%)
`With PMID`	5.75M (95.8%)
`With Abstract`	4.83M (80%)
`With Paragraph`	5.75M (95.8%)
`With References`	5.21M (86.8%)

Table 3: Statistics of the XML Parser output.

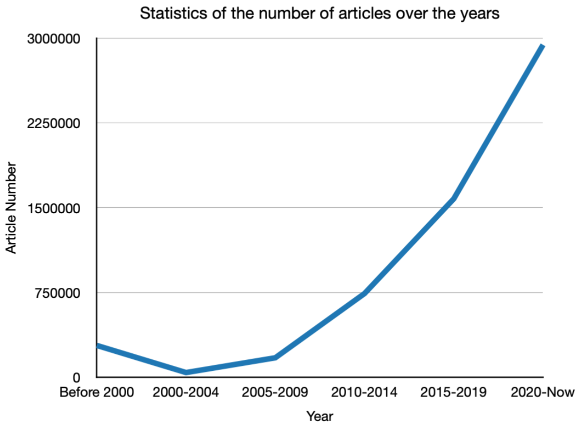


Figure 7: Statistic of the number of articles over the years in Europe PMC

abstracts and over 9.6 million full-text articles. Of these, nearly 6 million full-text open-access articles are available in XML format via the Europe PMC web services or FTP site.

The XML Parser in our pipeline is used to parse all the open-access articles mentioned above. The statistical results of the output of this module are presented in Table 3.

Next, the parsed text is processed by the finding identification module and the citation sentence extraction module. In terms of finding identification results, approximately 4.25 million articles have at least one finding. On average, we obtained 2.4 findings for each article that had findings, totaling around 10 million findings. For citation sentence extraction results, roughly 3.69 million articles have at least one citation sentence. We obtained 56 citation sentences on average for each article with citation sentences, amounting to approximately 200 million citation sentences in total. After dropping the citations not in Europe PMC, roughly 3.28 million articles have at least one citation sentence, with 20 citation sentences on average for each article and 67 million citation sentences in total.

Finally, the similarity scores were calculated based on these findings and citation sentences. These findings, citation sentences, and similarity scores are then utilized to create the finding-citation graph. We obtained 14.25 million nodes in total, consisting of 4.25 million article nodes and 10 million finding nodes. We got 77 million edges in total, of which there are 67 million edges representing citation relationships.

6 Discussion

6.1 Findings not in the abstract

From the literature review, it is clear that most previous studies primarily focus on identifying finding sentences from abstracts, often neglecting other sections. In our proposal to identify multiple findings, we are interested in determining how many findings are not included in the abstract, meaning the sentences containing these findings are not found in the abstract. From the 10 million findings we identified, we discovered that nearly 44% of the findings are not mentioned in the abstract. This percentage is slightly larger than expected. However, it aligns with our understanding that the abstract typically only describes the main findings, while other sections may discuss additional or side findings.

6.2 Distribution of similarity score between findings and citation sentences

The sentence embedding similarity score is utilized to measure how much information from the cited paper's findings is contained within the citation sentence. The larger it is, the more information it contains. The smaller it is, the less information it contains. Understanding the distribution of such similarity scores can help us determine how the findings are cited. The distribution can be seen in Figure 8. From the figure, it is clear that nearly half of the similarity scores are lower than 0.4. This suggests that half of the citation sentences contain less information about the findings of the cited papers. It meets our experience that most of the citations are used in the literature review section and for providing background information.

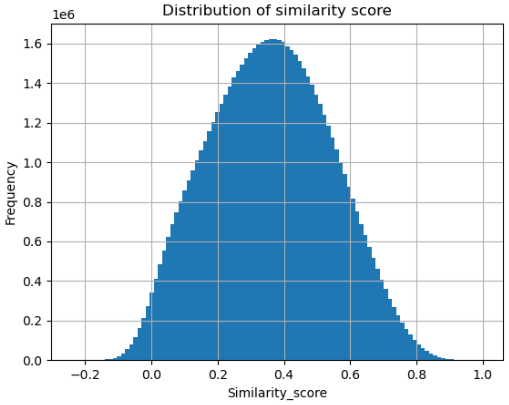


Figure 8: Distribution of the similarity score between citation sentences and cited findings

6.3 Limitations

Currently, our method only extracts sentences with the sentence marker for the citation, without considering the citation span. This approach might lead to some errors in matching the citation sentence with the findings.

Besides, the citation relations are based on the citation sentence and citation marker extraction, which would lead to error and loss to the graph.

Moreover, the citation graph is built using a closed dataset, specifically the Europe PubMed dataset. This approach excludes citations and articles not found in this dataset, inevitably leading to an incomplete graph.

The mentioned limitations above will help us identify areas where we can improve our graph optimization strategies in the future.

7 Conclusion

We introduce a new fine-grained citation graph, the finding-citation graph. Unlike the traditional citation graph which only contains papers as nodes, the finding-citation graph also includes the findings, representing the results of the academic papers. This graph facilitates more detailed studies at the finding level, such as evaluating findings and tracking the progression of research.

We also present a new pipeline for constructing this graph. This pipeline mainly consists of three modules: finding identification, citation sentence extraction, and graph construction. As there is no such pipeline to build the finding-citation graph, we design an evaluation to confirm the graph's quality. The finding identification module achieved 91% accuracy for finding sentence identification and 80% accuracy for multiple findings. The citation sentence extraction module got a 90% F1-score on the paper-level evaluation and 97.7% accuracy on the tuple-level evaluation. The outputs of the two modules are used to construct the graph and confirm its quality.

Finally, we built a finding-citation graph using Europe PMC. Our graph comprises 14.25 million nodes, with 4.25 million being academic papers and the rest being findings from those papers. It also includes 76 million edges, with 66 million representing citation relations.

The definition and creation of the FCG is an essential step for our future research. We plan to use it to assess research findings from a citation perspective and pinpoint future research directions at the finding level.

Ethics Statement

The paper considers the introduction of a new citation network, a finding-citation network, and a pipeline to construct such a graph. We did not work with limited datasets and only used open-source datasets.

Acknowledgements

This work was supported by the UKRI Biotechnology and Biology Sciences Research Council [BB/X511833/1], Digital Environment and Research Institute (DERI), the Queen Mary University of London, and Exscientia.

We thank Arkaitz Zubiaga and Daniel Crowther for their valuable feedback and suggestions on the project. We also thank the Semantic Scholar team for assisting with data access.

References

Titipat Achakulvisut, Daniel Acuna, and Konrad Kording. 2020a. Pubmed parser: A python parser for pubmed open-access xml subset and medline xml dataset xml dataset. *Journal of Open Source Software*, 5(46):1979.

Titipat Achakulvisut, Titipat Bhagavatula, Daniel Acuna, and Konrad Kording. 2020b. Claim extraction in biomedical publications using deep discourse model and transfer learning.

Steven Bird, Robert Dale, Bonnie Dorr, Bryan Gibson, Mark Joseph, Min-Yen Kan, Dongwon Lee, Brett Powley, Dragomir Radev, and Yee Fan Tan. 2008. The ACL Anthology reference corpus: A reference dataset for bibliographic research in computational linguistics. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Elliot Bolton, Abhinav Venigalla, Michihiro Yasunaga, David Hall, Betty Xiong, Tony Lee, Roxana Daneshjou, Jonathan Frankle, Percy Liang, Michael Carbin, and Christopher D. Manning. 2024. Biomedlm: A 2.7b parameter language model trained on biomedical text.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Arman Cohan, Waleed Ammar, Madeleine van Zuylen, and Field Cady. 2019. Structural scaffolds for citation intent classification in scientific publications. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3586–3596, Minneapolis, Minnesota. Association for Computational Linguistics.

Franck Dernoncourt and Ji Young Lee. 2017. PubMed 200k RCT: a dataset for sequential sentence classification in medical abstracts. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 308–313, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Li Fang, Qingyu Chen, Chih-Hsuan Wei, Zhiyong Lu, and Kai Wang. 2023. Bioformer: an efficient transformer language model for biomedical text mining.

Michael Färber, Alexander Thiemann, and Adam Jatowt. 2018a. A high-quality gold standard for citation-based tasks. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Michael Färber, Alexander Thiemann, and Adam Jatowt. 2018b. *To Cite, or Not to Cite? Detecting Citation Contexts in Text*, pages 598–603.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Dixon, Maciej Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu-hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. 2024. Gemma: Open models based on gemini research and technology.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare*, 3(1).

Katherine Gundolf and Matthias Filser. 2013. Management research and religion: A citation analysis. *Journal of Business Ethics*, 112:177–185.

Ivan Heibi, Silvio Peroni, and David Shotton. 2019. Software review: Coci, the opencitations index of crossref open doi-to-doi citations. *Scientometrics*, 121(2):1213–1228.

Christian Herzog, Daniel Hook, and Stacy Konkiel. 2020. Dimensions: Bringing down barriers between scientometricians and data. *Quantitative Science Studies*, 1:387–395.

Pradeep Kumar Hota, Balaji Subramanian, and Gopalakrishnan Narayanamurthy. 2020. Mapping the intellectual structure of social entrepreneurship research: A citation/co-citation analysis. *Journal of Business Ethics*, pages 1–26.

B. Ian Hutchins, Kirk L. Baker, Matthew T. Davis, Mario A. Diwersy, Ehsanul Haque, Robert M. Harriman, Travis A. Hoppe, Stephen A. Leicht, Payam Meyer, and George M. Santangelo. 2019. The nih open citation collection: A public access, broad coverage resource. *PLOS Biology*, 17(10):e3000385.

Rodney Michael Kinney, Chloe Anastasiades, Russell Authur, Iz Beltagy, Jonathan Bragg, Alexandra Buraczynski, Isabel Cachola, Stefan Candra, Yoganand Chandrasekhar, Arman Cohan, Miles Crawford, Doug Downey, Jason Dunkelberger, Oren Etzioni, Rob Evans, Sergey Feldman, Joseph Gorney, David W. Graham, F.Q. Hu, Regan Huff, Daniel King, Sebastian Kohlmeier, Bailey Kuehl, Michael Langan, Daniel Lin, Haokun Liu, Kyle Lo, Jaron Lochner, Kelsey MacMillan, Tyler Murray, Christopher Newell, Smita R Rao, Shaurya Rohatgi, Paul Sayre, Zejiang Shen, Amanpreet Singh, Luca Soldaini, Shivashankar Subramanian, A. Tanaka, Alex D Wade, Linda M. Wagner, Lucy Lu Wang, Christopher Wilhelm, Caroline Wu, Jiangjiang Yang, Angele Zamarron, Madeleine van Zuylen, and Daniel S. Weld. 2023. The semantic scholar open data platform. *ArXiv*, abs/2301.10140.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Zhentao Liang, Jin Mao, Kun Lu, and Gang Li. 2021. Finding citations for pubmed: a large-scale comparison between five freely available bibliographic data sources. *Scientometrics*, 126(12):9519–9542.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The semantic scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online. Association for Computational Linguistics.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Vinodkumar Prabhakaran, William L. Hamilton, Dan McFarland, and Dan Jurafsky. 2016. Predicting the rise and fall of scientific topics from trends in their rhetorical framing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1170–1180, Berlin, Germany. Association for Computational Linguistics.

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pretraining.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June (Paul) Hsu, and Kuansan Wang. 2015. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15 Companion, page 243–246, New York, NY, USA. Association for Computing Machinery.

Kazunari Sugiyama, Tarun Kumar, Min-Yen Kan, and Ramesh C. Tripathi. 2010. Identifying citing sentences in research papers using supervised learning. In *2010 International Conference on Information Retrieval Knowledge Management (CAMP)*, pages 67–72.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Xin Wei, Md Reshad UI Hoque, Jian Wu, and Jiang Li. 2023. Claimdistiller: Scientific claim extraction with supervised contrastive learning. 3451:65–77.

Dustin Wright and Isabelle Augenstein. 2021. Cite-Worth: Cite-worthiness detection for improved scientific document understanding. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1796–1807, Online. Association for Computational Linguistics.

Dustin Wright, Jiaxin Pei, David Jurgens, and Isabelle Augenstein. 2022. Modeling information change in science communication with semantically matched paraphrases. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1783–1807, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2022. LinkBERT: Pretraining language models with document links. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8003–8016, Dublin, Ireland. Association for Computational Linguistics.

Weiping Yue and Concepción S. Wilson. 2004. Measuring the citation impact of research journals in clinical neurology: A structural equation modelling analysis. *Scientometrics*, 60(3):317–332.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

Qihang Zhao. 2020. Utilizing citation network structure to predict citation counts: A deep learning approach.

Evaluating Large Language Models for Predicting Protein Behavior under Radiation Exposure and Disease Conditions

Ryan Engel
Stony Brook University
ryengel@cs.stonybrook.edu

Gilchan Park
Brookhaven National Laboratory
gpark@bnl.gov

Abstract

The primary concern with exposure to ionizing radiation is the risk of developing diseases. While high doses of radiation can cause immediate damage leading to cancer, the effects of low-dose radiation (LDR) are less clear and more controversial. To further investigate this, it necessitates focusing on the underlying biological structures affected by radiation. Recent work has shown that Large Language Models (LLMs) can effectively predict protein structures and other biological properties. The aim of this research is to utilize open-source LLMs, such as Mistral, Llama 2, and Llama 3, to predict both radiation-induced alterations in proteins and the dynamics of protein-protein interactions (PPIs) within the presence of specific diseases. We show that fine-tuning these models yields state-of-the-art performance for predicting protein interactions in the context of neurodegenerative diseases, metabolic disorders, and cancer. Our findings contribute to the ongoing efforts to understand the complex relationships between radiation exposure and disease mechanisms, illustrating the nuanced capabilities and limitations of current computational models. The code and data are available at: https://github.com/Rengel2001/SURP_2024

1 Introduction

The exploration of the biological consequences of ionizing radiation on human health has long been a focal point of medical and environmental research. High doses of radiation are linked to immediate cellular damage and an increased risk of cancer (Wang et al., 2018). However, the implications of low-dose radiation (LDR) exposure remain a topic of significant debate. Emerging evidence suggests potential associations with various non-cancerous diseases, including neurodegenerative and cardiovascular diseases (Sharma et al., 2018; Kamiya et al., 2015). Additionally, others show that cancer is a result of low-dose radiation exposure (Shah et al., 2014; Hauptmann et al., 2020).

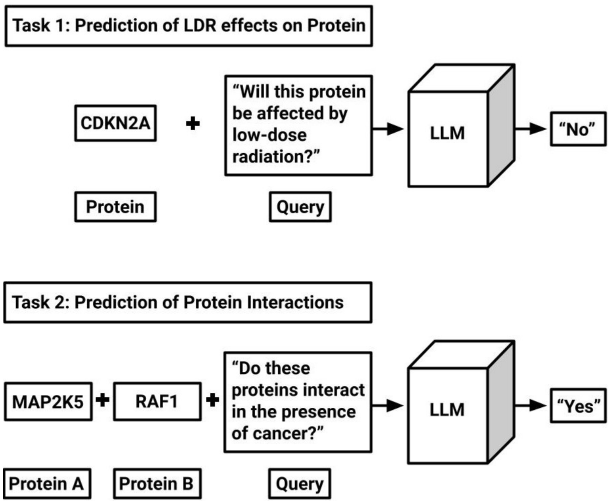

Task 1: Prediction of LDR effects on Protein
CDKN2A
+
"Will this protein be affected by low-dose radiation?"
LLM
"No"
Protein
Query
Task 2: Prediction of Protein Interactions
MAP2K5
+
RAF1
+
"Do these proteins interact in the presence of cancer?"
LLM
"Yes"
Protein A
Protein B
Query

Figure 1: Tasks Utilizing LLMs for Protein Behavior Prediction.

Understanding these effects at the molecular level, particularly in relation to protein structure and function, is crucial for developing protective measures. Similarly, protein-protein interactions (PPIs) are vital for various cellular processes and play a critical role in understanding disease mechanisms. Furthermore, there exists extensive PPI data, compiled into comprehensive public databases like BioGRID (Oughtred et al., 2021), STRING (Alanis-Lobato et al., 2016), HIPPIE (Szklarczyk et al., 2021), and Kegg (Kanehisa et al., 2017). Considerable research has been dedicated to understanding general protein interactions; however, there is a lack of studies examining protein interaction networks in the context of specific diseases.

The overarching goal of this research is to determine the efficacy of LLMs in accurately predicting complex biological processes related to protein function under various conditions. We employ three state-of-the-art LLMs, to analyze data from six diverse datasets. These datasets represent

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 427–439
August 16, 2024. ©2024 Association for Computational Linguistics

two distinct categories. The first focuses on the effects of LDR on proteins, and the second highlights the PPI network present within specific diseases. We formalize this data into two binary classification tasks, which are illustrated in Figure 1. This approach not only demonstrates the versatility of LLMs in biological research but also paves the way for novel insights into the molecular dynamics influenced by radiation exposure and disease processes. Our contributions in this paper include:

1. Organizing 6 key datasets, which are then split into 13 subsets, each designed to emphasize different experimental conditions.
2. Conducting a comprehensive evaluation of three open-source LLMs, comparing the performance of pre-trained models with the fine-tuned models.
3. Investigating the level of knowledge that LLMs have regarding protein behaviors and reviewing their current limitations for these tasks.
4. Analyzing the proteins that occur in both the LDR datasets and the PPI datasets, to highlight which proteins in each network are significantly deregulated by radiation exposure.

2 Related Works

2.1 Low-Dose Radiation Research

There has been a great deal of research focused on the effects of radiation on biological systems. Many studies exploring the field use traditional methods and there has been significant progress (Khan and Wang, 2022; Tatjana Paunesku and Woloschak, 2021; Ji et al., 2019). However, the application of machine learning to these studies has been limited. Notably, one approach employed artificial neural networks (ANNs) within the Rosetta suite to predict protein post-translational modifications (PTMs) relevant to radiation-induced effects (Ertelt et al., 2024). Another study used machine learning to identify potential methionine oxidation sites, a modification also associated with oxidative stress from radiation (Aledo et al., 2017). These instances showcase the emerging intersection of computational power with radiation biology research.

2.2 PPI Prediction Methods

The abundance of PPI data has prompted significant advancements in molecular biology research. Recently, computational techniques employing machine learning and graph embeddings have been developed for PPI prediction. One approach employs Graph-BERT, ProtBERT, and SeqVec models within a PPI network graph, showcasing the efficacy of language models (Jha et al., 2023). Another emerging trend is the use of Convolutional Neural Networks (CNNs), with studies employing Bio2Vec coupled with CNNs to predict PPIs from sequences (Wang et al., 2019; Hashemifar et al., 2018). The PIPR method simplifies PPI prediction by using sequence data alone, surpassing many traditional models in both basic and complex PPI tasks (Chen et al., 2019). While many methods focus on general PPI prediction, the NECARE model (Qiu et al., 2021) excels at predicting cancer-associated PPIs using a deep learning framework with a Relational Graph Convolutional Network (R-GCN). Similarly, the symmetric logistic matrix factorization (symLMF) approach (Pei et al., 2021) accurately predicts PPIs, including those involved in neurodegenerative and metabolic disorders, outperforming most classifiers.

2.3 Language Models for Molecular Biology

Concurrently, advancements in computational biology have also leveraged language models and the transformer architecture (Vaswani et al., 2017) to achieve significant breakthroughs in biomolecular and proteomics research. At the forefront, AlphaFold (Jumper et al., 2021) has set a precedent by employing innovative deep learning techniques to predict protein structures with remarkable accuracy. Building upon these foundations, protein language models like ProGen2 (Nijkamp et al., 2022), ProGPT2 (Ferruz et al., 2022), and ProLlama (Lv et al., 2024), have further developed the applications of language modelling for proteomics. Additionally, this has led to advancements in general purpose biological language models like BioGPT (Luo et al., 2022), and BioMedLM (Bolton et al., 2024).

2.4 General Purpose LLMs

Large-scale language models like Llama (Touvron et al., 2023a), and its subsequent iterations including Llama 2 (Touvron et al., 2023b), Llama 3 (AI@Meta, 2024) and Alpaca (Taori et al., 2023), have highlighted the importance of data design and task-specific training in improving model performance across a variety of tasks. Additionally, the creation of the Mistral (Jiang et al., 2023) model helps to bring open-source LLMs to the forefront of scientific innovation. These strides in LLM research have brought significant advancements in other scientific disciplines (Zhang et al., 2024). We aim to utilize such LLMs to further advance research on LDR exposure and to analyze how this might affect protein networks and specific diseases.

3 LLMs and Datasets

In this study, we employ three open-source LLMs, Mistral (7B), Llama 2 (7B), and Llama 3 (8B), to investigate two primary areas of biological research: the effects of low-dose radiation (LDR) on proteins and the dynamics of PPIs in the context of specific diseases. These models were chosen because of their state-of-the-art performance in many natural language processing (NLP) tasks. Additionally, their open-source nature allows for broad accessibility and modification by researchers across disciplines, which promotes transparency and collaborative advancements in both NLP and other scientific domains.

To facilitate a comprehensive analysis, our methodology encompasses six core datasets, which are further subdivided into 13 distinct subsets based on specific experimental parameters and objectives. The first 3 datasets primarily explore the effects of LDR on protein deregulation. These 3 sets are further divided into 10 subsets, emphasizing different experimental conditions. The last 3 datasets focus on PPIs in the presence of specific diseases, namely neurodegenerative, metabolic, and cancer.

The subsets of the LDR data are much smaller than the PPI datasets, which is why these were combined into dataset 3c. Dataset 3c's larger size is shown in comparison with the other datasets in Figure 2. The details about each dataset is outlined in Appendix A.

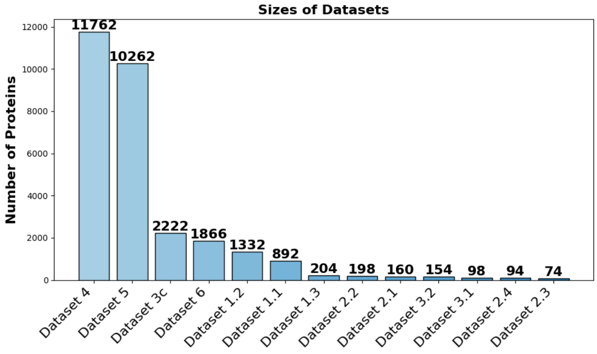


Figure 2: Comparison of Dataset Sizes

4 Experiments

The methodology for analyzing each dataset began with data pre-processing, executed through a Python script tailored to appropriately structure the raw data. Subsequently, this processed data was used to create prompts that fit the prompting strategies outlined in Appendix A. These prompts were then saved in a JSON file, and were subsequently used as input to the LLMs.

For deploying the models, a separate Python script using the Hugging Face Transformers library loaded the models onto 4×NVIDIA A100 80GB GPUs. These pre-trained models were then presented with the JSON file prompts and the performance of each model was recorded.

4.1 Experimental Setup

Our experimental setup across the datasets implemented a binary classification task, instructing the models to produce a "yes" or "no" answer in response to each prompt. The generated responses from each model necessitate the deployment of an algorithm to parse these outputs effectively. If the given string "yes" or "no" is not found in the model's response, this response is marked as the opposite of the true label. This is a result of using causal language models, which are designed for text generation. To optimize this task, the “Data Collator for Completion-Only Language Models“ and the SFT (Supervised fine-tuning) Trainer from the Hugging Face library were utilized in training the models to give the correct response structure.

4.2 Data Split

We structured the training process differently for each of the two tasks. For the LDR task, we divided the prompts for each dataset into an 80/10/10 split for training, validation, and testing, respectively. The PPI datasets 4 and 5 utilized a 5-fold cross validation setup, where 4 sets were used for training and 1 set was used for testing in each fold. Similarly, the PPI dataset 6 used a 5-fold cross validation setup but instead 3 sets were used for training, 1 set for validation, and 1 set for testing. This was carried out to replicate the experimental conditions used in the benchmark models.

4.3 Fine-Tuning

During the training process, we employed Parameter Efficient Fine-Tuning (PEFT) (Mangrulkar et al., 2022), a method focused on selectively modifying a subset of the model's parameters rather than the entire set. Low-Rank Adaptation (LoRA) (Hu et al., 2021) is a specialized PEFT technique that was utilized when fine-tuning the LLMs for these tasks. Additionally, we used QLoRA (Dettmers et al., 2023) to reduce the GPU memory required for training Llama 3 on datasets 4 and 5. This approach was essential because the combined size of these datasets and the 8 billion parameter model required more efficient memory usage than traditional LoRA.

5 Results

Every phase of the model training process was documented and analyzed. The evaluation metrics used include accuracy, Matthews Correlation Coefficient (MCC), specificity, macro precision, and macro F1 Score.

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	0.367	-0.343	0.068	0.290	0.304
Llama 2 (3-shot)	0.556	0.110	0.386	0.558	0.541
Llama 3 (3-shot)	0.489	0.0	1.0	0.244	0.328
Mistral (LoRA)	0.500	0.058	**0.977**	0.580	0.369
Llama 2 (LoRA)	0.522	0.061	0.750	0.534	0.500
Llama 3 (LoRA)	**0.567**	**0.155**	0.773	**0.585**	**0.551**

Table 1: Performance Comparison for Dataset 1.1

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	0.291	-0.461	0.027	0.210	0.239
Llama 2 (3-shot)	**0.567**	**0.153**	0.479	**0.578**	**0.566**
Llama 3 (3-shot)	0.545	0.0	**1.0**	0.272	0.353
Mistral (LoRA)	0.493	0.007	0.384	0.503	0.490
Llama 2 (LoRA)	0.537	-0.006	0.932	0.494	0.401
Llama 3 (LoRA)	0.552	0.054	0.945	0.554	0.420

Table 2: Performance Comparison for Dataset 1.2

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	0.286	-0.230	0.133	0.368	0.279
Llama 2 (3-shot)	0.381	-0.067	0.267	0.467	0.381
Llama 3 (3-shot)	**0.714**	0.0	**1.0**	0.357	0.417
Mistral (LoRA)	0.381	-0.241	0.400	0.391	0.358
Llama 2 (LoRA)	0.381	-0.447	**0.533**	0.286	0.276
Llama 3 (LoRA)	0.571	**0.279**	0.467	**0.630**	**0.568**

Table 3: Performance Comparison for Dataset 1.3

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	0.125	-0.745	0.0	0.083	0.111
Llama 2 (3-shot)	0.438	-0.035	0.3	0.482	0.435
Llama 3 (3-shot)	0.625	0.0	**1.0**	0.313	0.385
Mistral (LoRA)	0.688	0.313	0.8	0.664	0.654
Llama 2 (LoRA)	0.688	0.423	0.6	0.706	0.686
Llama 3 (LoRA)	**0.813**	**0.592**	**0.9**	**0.809**	**0.792**

Table 4: Performance Comparison for Dataset 2.1

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	0.095	-0.767	0.0	0.059	0.087
Llama 2 (3-shot)	0.524	**0.224**	0.4	**0.607**	**0.523**
Llama 3 (3-shot)	**0.714**	0.0	**1.0**	0.357	0.417
Mistral (LoRA)	**0.714**	0.0	**1.0**	0.357	0.417
Llama 2 (LoRA)	0.286	0.0	0.0	0.143	0.222
Llama 3 (LoRA)	0.524	-0.167	0.667	0.417	0.417

Table 5: Performance Comparison for Dataset 2.2

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	0.25	-0.5	0.25	0.25	0.25
Llama 2 (3-shot)	0.375	-0.378	0.0	0.214	0.273
Llama 3 (3-shot)	0.5	0	**1.0**	0.25	0.333
Mistral (LoRA)	**0.625**	**0.258**	0.5	**0.633**	**0.619**
Llama 2 (LoRA)	**0.625**	**0.258**	0.75	**0.633**	**0.619**
Llama 3 (LoRA)	0.5	0	**1.0**	0.25	0.333

Table 6: Performance Comparison for Dataset 2.3

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	0.1	-0.816	0.167	0.1	0.091
Llama 2 (3-shot)	0.4	-0.102	0.167	**0.438**	**0.375**
Llama 3 (3-shot)	**0.6**	**0.0**	**1.0**	0.3	**0.375**
Mistral (LoRA)	**0.6**	**0.0**	**1.0**	0.3	**0.375**
Llama 2 (LoRA)	0.4	**0.0**	0.0	0.2	0.286
Llama 3 (LoRA)	0.4	**0.0**	0.0	0.2	0.286

Table 7: Performance Comparison for Dataset 2.4

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	0.364	**0.0**	0.0	0.182	0.267
Llama 2 (3-shot)	0.364	-0.463	0.571	0.25	0.267
Llama 3 (3-shot)	**0.636**	**0.0**	**1.0**	**0.318**	**0.389**
Mistral (LoRA)	0.364	**0.0**	0.0	0.182	0.267
Llama 2 (LoRA)	0.364	**0.0**	0.0	0.182	0.267
Llama 3 (LoRA)	0.273	-0.418	0.0	0.15	0.214

Table 8: Performance Comparison for Dataset 3.1

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	0.438	0.0	0.0	0.219	0.304
Llama 2 (3-shot)	0.438	-0.098	0.333	0.450	0.435
Llama 3 (3-shot)	**0.625**	**0.293**	**1.0**	**0.8**	**0.5**
Mistral (LoRA)	0.563	0.0	**1.0**	0.281	0.360
Llama 2 (LoRA)	0.438	0.0	0.0	0.219	0.304
Llama 3 (LoRA)	0.563	0.0	**1.0**	0.281	0.360

Table 9: Performance Comparison for Dataset 3.2

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	0.247	-0.593	0.0	0.173	0.198
Llama 2 (3-shot)	0.475	-0.015	**0.769**	0.491	0.443
Llama 3 (3-shot)	0.493	-0.038	0.317	0.480	0.473
Mistral (LoRA)	**0.552**	**0.090**	0.423	0.546	**0.540**
Llama 2 (LoRA)	0.547	0.066	0.154	**0.549**	0.459
Llama 3 (LoRA)	0.516	0.014	0.375	0.507	0.502

Table 10: Performance Comparison for Dataset 3c

Tables 1-13 indicate the performance of both the pre-trained models, and their fine-tuned counterparts on each of the 13 datasets. We evaluated the pre-trained models using the same procedure as the fine-tuned models, the only difference is that the

Model	Acc. (%)	MCC (%)	Spec. (%)	Prec.(%)	F1 (%)
Mistral (3-shot)	38.44±0.46	-31.79±0.54	4.15±0.16	28.16±0.28	30.23±0.25
Llama 2 (3-shot)	55.14±0.16	13.18±0.29	23.79±0.57	58.47±0.23	50.23±0.17
Llama 3 (3-shot)	50.38±0.36	6.17±0.27	**1.0±0.0**	75.10±0.18	34.18±0.17
Mistral (LoRA)	62.34±6.88	25.53±14.37	97.89±1.17	48.49±12.84	51.97±10.36
Llama 2 (LoRA)	87.28±0.41	76.63±1.03	88.59±0.95	87.33±0.22	87.28±0.31
Llama 3 (QLoRA)	**88.27±1.08**	**76.92±2.12**	92.81±1.06	**88.58±1.08**	**88.26±1.07**
SymLMF (Reported)	86.11±1.05	74.29±2.07	N/A	83.24±1.28	N/A

Table 11: Performance Comparison for Dataset 4

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	35.52±0.76	-39.08±0.82	1.98±0.14	23.65±0.56	27.33±0.48
Llama 2 (3-shot)	51.45±0.81	6.66±1.16	6.42±0.37	57.66±1.31	39.09±0.57
Llama 3 (3-shot)	56.03±0.21	12.12±0.46	53.69±0.34	56.06±0.23	56.00±0.21
Mistral (LoRA)	62.18±6.61	25.08±13.33	**93.80±3.89**	57.70±11.91	51.93±10.16
Llama 2 (LoRA)	84.63±0.24	69.20±0.66	84.06±0.92	84.51±0.34	84.43±0.34
Llama 3 (QLoRA)	**91.28±0.87**	**82.57±1.73**	90.41±1.08	**91.29±.86**	**91.28±0.87**
SymLMF (Reported)	81.37±1.04	63.31±2.07	N/A	77.70±1.07	N/A

Table 12: Performance Comparison for Dataset 5

Model	Acc.	MCC	Spec.	Prec.	F1
Mistral (3-shot)	41.91±1.19	-23.81±1.55	5.15±0.43	32.45±0.95	32.79±0.69
Llama 2 (3-shot)	57.61±0.91	16.69±2.40	39.18±1.42	59.0±1.32	56.09±1.0
Llama 3 (3-shot)	53.96±1.62	16.40±2.75	**97.83±0.55**	67.25±2.83	42.91±1.68
Mistral (LoRA)	83.76±8.12	68.81±15.40	93.30±2.15	79.20±12.41	80.69±10.85
Llama 2 (LoRA)	93.25±0.84	86.84±1.49	93.39±1.38	93.55±0.74	93.22±0.84
Llama 3 (LoRA)	**93.94±0.25**	**88.04±0.47**	91.83±1.12	**94.09±0.22**	**93.92±0.25**
NECARE (Reported)	N/A	84.0±3.0	92.0±2.0	90.0±2.0	90.0±2.0

Table 13: Performance Comparison for Dataset 6

models were prompted with example questions or “shots“ before the dataset prompt was given. The term "3-shot" refers to the 3 example questions prompted before the dataset's prompt. The results of these experiments demonstrated that LLMs were particularly effective when fine-tuned on larger, well-structured datasets, as evidenced by their success in the PPI prediction task.

5.1 Performance

When fine-tuned with QLoRA, Llama 3 shows superior performance on the PPI prediction task for each of the three datasets. On the neurodegenerative (Table 11) and metabolic disorder (Table 12) PPI prediction tasks, it scores an accuracy of 88.27% and 91.28% respectively. These values outperform the current best model SymLMF (Pei et al., 2021), which achieves only 86.11% and 81.37%.

Furthermore, this model fine-tuned with LoRA achieves a precision of 96.9%, which outperforms the 94% precision achieved with NECARE (Qiu et al., 2021) as shown in table 13. It is clear that the fine-tuned Llama 3 model is currently the best prediction method for identifying PPIs in the presence neurodegenerative diseases, metabolic disorders, and cancer.

5.2 Discussion

We show that fine-tuning the LLMs can increase performance by a substantial margin. However, this depends heavily on the size of the dataset used to train the model, and the specific prompting techniques used. While fine-tuning significantly boosted accuracy in datasets 4, 5, and 6 by up to 50%, model performance on datasets 1, 2, and 3 exhibited less pronounced improvements after fine-tuning (Tables 1-10).

In analyzing the discrepancies in model performance between the PPI and LDR tasks, one notable difference lies in the composition of the prompts used for each task. For the PPI task, each prompt includes two variable protein names. This dual-protein structure of the prompts likely provides the model with a relational context that aids in discerning interaction patterns between proteins, facilitating more effective learning and prediction.

In contrast, the LDR task prompts feature only one variable protein name, potentially limiting the model's learning and predictive capabilities due to insufficient relational or comparative data. The single protein name reduces the available contextual cues for predicting deregulation. This prompt design likely contributes to the lower accuracy in the LDR task, as the model may struggle to infer the broader biological impacts of LDR exposure from a solitary protein reference.

These results not only illustrate the current constraints of these models but also suggests potential avenues for improvements, such as the development of more domain-specific datasets related to LDR, or the application of prompt engineering techniques.

6 Evaluation of Model Predictions

In this section, we analyze the predictions made by the LLMs and highlight some of the proteins that were correctly and incorrectly identified. We focus on interpreting the results obtained from our experiments in tables 1-13, examining the predictions made and identifying patterns in each model's output to understand their current limitations.

6.1 Correctly Identified Proteins

After analyzing the model output for each LLM, there are a few commonalities between the correctly identified proteins. Many of the names follow standard naming conventions in molecular biology, such as using abbreviations or acronyms that represent the function or family of the protein. Some examples include: SLC (Solute Carrier) proteins slc9a6, slc3a2, slc27a4, slc1a1, slc38a3, and RP (Ribosomal Protein) proteins rpl24, rpl22, rpl9, rpl15, rps11, rps25, rps13, rps27rt.

Additionally, the proteins correctly identified seem to belong to various functional categories, such as cytoskeletal proteins: tubb4a, tubb, actb, signaling proteins: hras, gsk3b, camk2a, camk4, rab3b, and metabolic enzymes: aldh3b1, aldh1l1, psat1, cpt2, pnpo, ak5, pgm3.

Overall, the correctly identified proteins cover a diverse range of cellular functions, including signaling, metabolism, transport, cytoskeletal organization, and many others. The naming conventions and functional hints within the protein names suggest that these proteins are well-studied and recognized by the models, potentially due to their importance in various biological processes and their prevalence in scientific literature.

6.2 Incorrectly Identified Proteins

When contrasting the incorrectly identified proteins with the correctly identified ones, a few key differences can be observed. Specifically, the incorrectly identified protein names seem to follow less standardized naming conventions compared to the correctly identified ones. They lack common abbreviations or acronyms that indicate their functional categories or protein families.

Furthermore, it is more challenging to infer the functional categories or processes that the incorrectly identified proteins are involved in based solely on their names. These proteins could be less well-known or less extensively researched, making it more challenging for the models to accurately identify them.

Additionally, LLMs might have biases or limitations in their training data or algorithms, which could contribute to the discrepancies in identification accuracy. Ultimately, the correctly identified proteins seem to follow more recognizable naming conventions, belong to well-characterized functional categories, and potentially have a more substantial presence in scientific literature, which could explain why they were more accurately identified by the models compared to the incorrectly identified ones.

7 Dataset Cross-Reference Analysis

Independent of the LLM experiments, we conduct a dataset cross-reference analysis to identify the common proteins between the LDR and PPI datasets, highlighting those that may be involved in both processes. Through this extensive analysis, we gained a deeper understanding of the data utilized for training these LLMs and enhanced our understanding of the protein dynamics involved in both radiation response and disease mechanisms.

We identified overlaps between the PPI datasets 4, 5, and 6, and the combined LDR dataset 3c. The positive interaction pairs were identified for each of datasets 4 (11,762 proteins), 5 (10,262 proteins), and 6 (1,866 proteins). Subsequently, the significantly affected proteins in the combined dataset 3c were identified (1,111 proteins). Our findings show that the highest percentage of overlap with the LDR data was with dataset 4, the neurodegenerative PPI dataset.

7.1 Dataset Analysis Metrics

The metrics used for these experiments include the percentage of overlap, the multiset coverage, the Jaccard index, and the weighted Jaccard index. The difference between the percentage overlap and the multiset coverage is that the multiset coverage takes into account the frequency of reoccurring proteins between all interactions. In other words, multiset coverage includes duplicate protein names, where the percentage overlap uses only unique proteins names.

The reasoning for calculating both multiset coverage and percent overlap is because if a specific protein occurs frequently in the protein interaction network, it likely contributes more to the overall biological structure. Thus, including the duplicate proteins in the calculation of multiset coverage illustrates the extent to which these proteins affect the network.

Additionally, the Jaccard index was used for calculating the set similarity for the unique proteins, and the weighted Jaccard index was used when accounting for duplicate proteins. These values measure the similarity between the two sets, while accounting for their sizes through normalization.

7.2 Neurodegenerative Diseases PPI

The neurodegenerative diseases PPI dataset exhibited the highest percentage of unique protein overlap (14.02%) and multiset coverage (22.21%). The Jaccard Index for unique proteins was 0.0633 and the Weighted Jaccard Index was 0.2546, indicating a significant shared profile. This neurodegenerative PPI dataset contains 820 unique proteins in the PPI network. There were 115 unique proteins identified to overlap between the LDR data and PPI data. Some of these proteins include MAPT (Microtubule-Associated Protein Tau) (Medeiros et al., 2011), HTT (Huntingtin) (Tabrizi et al., 2019; Jimenez-Sanchez et al., 2017), APP (Amyloid Precursor Protein) (de la Vega et al., 2021; X et al., 2021), and GFAP (Glial Fibrillary Acidic Protein) (Yang and Wang, 2015; Kunchok et al., 2019), each of which have been shown to be linked with neurodegenerative diseases.

7.3 Metabolic Disorders PPI

The metabolic disorders PPI dataset showed a 7.14% overlap with unique proteins, and a multiset coverage of 13.78%. Both the Jaccard Index (0.0357) and Weighted Jaccard Index (0.1420) were lower compared to the neurodegenerative dataset, indicating less similarity with the LDR dataset but still notable overlap. This metabolic diseases PPI dataset contains 1036 unique proteins in the network. There were 74 unique proteins identified between both sets. Some of these proteins include ALDH2 (Aldehyde Dehydrogenase 2) (Wang et al., 2021; Chen et al., 2022), ACE (Angiotensin-Converting Enzyme) (Fountain et al., 2024), and ACAD8 (Acyl-CoA Dehydrogenase 8) (Zhuang et al., 2022), which have been shown to link to metabolic disorders.

7.4 Cancer PPI

The overlap in the cancer PPI dataset was more modest, with an 8.84% overlap and 4.72% multiset coverage, highlighting 19 unique overlapping proteins. The Jaccard Index was notably low at 0.0145, and the Weighted Jaccard Index at 0.0305. Some notable proteins identified include PAK1 (P21-Activated Kinase 1) (Belli et al., 2023), GRM1 (Glutamate Metabotropic Receptor 1) (Mehta et al., 2013; Nord et al., 2014), ANK1 (Ankyrin 1) (Tessema et al., 2017), and PTEN (Phosphatase and Tensin Homolog) (Liu et al., 2015). These proteins are illustrated in Figure 3. The highlighted proteins are also found in the combined LDR dataset, indicating that these proteins are significantly deregulated after exposure to LDR.

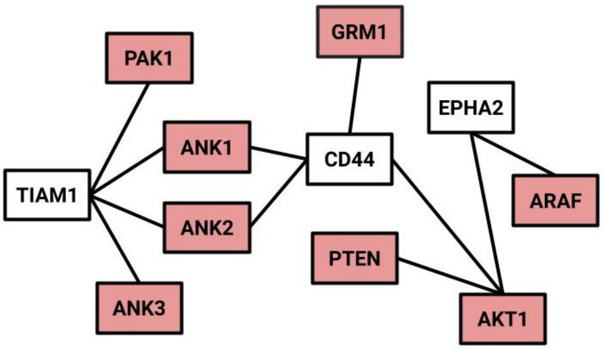


Figure 3: Cancer Protein Interaction Network. Highlighted Proteins are Significantly Affected by LDR.

7.5 Comparison

The higher overlap and Jaccard indices for dataset 4 show that there are more proteins in this network that are affected by LDR compared to those in the metabolic and cancer datasets. Similarly, the overlap of unique proteins between dataset 3c and dataset 6 is more than the overlap between datasets 3c and 5 despite its significantly larger size. This data suggests a higher probability that LDR affects cancer when compared to metabolic disorders. By highlighting the specific proteins overlapping between these datasets, we have identified key points for future research that can help bridge the gap between LDR exposure and disease mechanisms.

8 Conclusion

This study presents an exploration of the capabilities of LLMs in predicting the molecular dynamics of proteins under various conditions. By employing three state-of-the-art LLMs across multiple datasets, our research offers valuable insights into the potential utility and limitations of computational models for these tasks.

The fine-tuning process using LoRA proved to be a pivotal factor in enhancing model performance, demonstrating notable improvements in accuracy and predictive capabilities. Improving the accuracy of these models is key, because a major limitation of LLMs is their tendency to hallucinate, or give false information. Utilizing parameter efficient fine-tuning strategies helps to alleviate this problem while also maintaining an efficient computational complexity. Through the use of PEFT, the Llama 3 model outperforms the current best models for the PPI prediction tasks, indicating its potential for future advancements in biomolecular research.

Our analysis of protein identification by these models revealed intriguing patterns. Correctly identified proteins often belonged to well-characterized functional categories and were represented by standard naming conventions, suggesting that the pre-training on extensive biomedical literature may have equipped the models with a robust foundation of biological knowledge. Conversely, proteins that were incorrectly identified typically lacked these characteristics, possibly indicating areas where LLMs could benefit from further training or more focused dataset enrichment.

The cross-referencing of proteins affected by LDR with those involved in PPIs of neurodegenerative, metabolic, and cancer-related processes brought forth specific proteins that could be further explored in future studies. Notably, the neurodegenerative PPI dataset showed the highest overlap, where 115 unique proteins were identified in both datasets. These results highlight exactly which proteins in the PPI networks are significantly deregulated after LDR exposure, which could help to advance our understanding of how LDR affects disease mechanisms.

In conclusion, the integration of LLMs into biological research, particularly using fine-tuning techniques like LoRA, holds promising potential for advancing our understanding of the molecular mechanisms underpinning disease and radiation exposure. The versatility and scalability of these models make them instrumental tools in the ongoing quest to decode complex biological data. Their capacity to learn patterns and generate insights from extensive datasets holds immense promise for future research endeavors. Future work should focus on expanding the datasets, specifically the LDR data, and refining model architectures to further enhance the precision and applicability of LLMs in scientific discovery.

Acknowledgments

This material is based upon work for the LUCID: Low-dose Understanding, Cellular Insights, and Molecular Discoveries program supported by the U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research, under Contract DE-AC02-06CH11357.

References

AI@Meta. 2024. Llama 3 model card.

Gregorio Alanis-Lobato, Miguel A. Andrade-Navarro, and Martin H. Schaefer. 2016. HIPPIE v2.0: enhancing meaningfulness and reliability of protein–protein interaction networks. *Nucleic Acids Research*, 45(D1):D408–D414.

Juan C. Aledo, Francisco R. Cantón, and Francisco J. Veredas. 2017. A machine learning approach for predicting methionine oxidation sites. *BMC Bioinformatics*, 18(1):430.

Z Barjaktarovic, J Merl-Pham, O Azimzadeh, S J. Kempf, K Raj, M J. Atkinson, and S Tapio. 2017. Low-dose radiation differentially regulates protein acetylation and histone deacetylase expression in human coronary artery endothelial cells. *International Journal of Radiation Biology*, 93(2):156–164. PMID: 27653672.

Stefania Belli, Daniela Esposito, Alessandra Allotta, Alberto Servetto, Paola Ciciola, Ada Pesapane, Claudia M. Ascione, Fabiana Napolitano, Concetta Di Mauro, Elena Vigliar, Antonino Iaccarino, Carmine De Angelis, Roberto Bianco, and Luigi Formisano. 2023. Pak1 pathway hyper-activation mediates resistance to endocrine therapy and cdk4/6 inhibitors in er+ breast cancer. *npj Breast Cancer*, 9:48.

Elliot Bolton, Abhinav Venigalla, Michihiro Yasunaga, David Hall, Betty Xiong, Tony Lee, Roxana Daneshjou, Jonathan Frankle, Percy Liang, Michael Carbin, and Christopher D. Manning. 2024. Biomedlm: A 2.7b parameter language model trained on biomedical text. *Preprint*, arXiv:2403.18421.

C. H. Chen, B. R. Kraemer, and D. Mochly-Rosen. 2022. Aldh2 variance in disease and populations. *Dis Model Mech*, 15(6):dmm049601.

Muhao Chen, Chelsea J T Ju, Guangyu Zhou, Xuelu Chen, Tianran Zhang, Kai-Wei Chang, Carlo Zaniolo, and Wei Wang. 2019. Multifaceted protein–protein interaction prediction based on Siamese residual RCNN. *Bioinformatics*, 35(14):i305–i314.

M. Pagnon de la Vega, V. Giedraitis, W. Michno, L. Kilander, G. Güner, M. Zielinski, M. Löwenmark, R. Brundin, T. Danfors, L. Söderberg, I. Alafuzoff, L. N. G. Nilsson, A. Erlandsson, D. Willbold, S. A. Müller, G. F. Schröder, J. Hanrieder, S. F. Lichtenthaler, L. Lannfelt, D. Sehlin, and M. Ingelsson. 2021. The uppsala app deletion causes early onset autosomal dominant alzheimer's disease by altering app processing and increasing amyloid β fibril formation. *Cold Spring Harbor Perspectives in Medicine*, 7(7):a024240.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *Preprint*, arXiv:2305.14314.

Moritz Ertelt, Vikram Khipple Mulligan, Jack B. Maguire, Sergey Lyskov, Rocco Moretti, Torben Schiffner, Jens Meiler, and Clara T. Schoeder. 2024. Combining machine learning with structure-based protein design to predict and engineer post-translational modifications of proteins. *PLOS Computational Biology*, 20(3):1–20.

Noelia Ferruz, Steffen Schmidt, and Birte Höcker. 2022. Protgpt2 is a deep unsupervised language model for protein design. *Nature Communications*, 13(1):4348.

J. H. Fountain, J. Kaur, and S. L. Lappin. 2024. Physiology, renin angiotensin system. *StatPearls*.

Somaye Hashemifar, Behnam Neyshabur, Aly A Khan, and Jinbo Xu. 2018. Predicting protein–protein interactions through sequence-based deep learning. *Bioinformatics*, 34(17):i802–i810.

M. Hauptmann, R. D. Daniels, E. Cardis, H. M. Cullings, G. Kendall, D. Laurier, M. S. Linet, M. P. Little, J. H. Lubin, D. L. Preston, D. B. Richardson, D. O. Stram, I. Thierry-Chef, M. K. Schubauer-Berigan, E. S. Gilbert, and A. Berrington de Gonzalez. 2020. Epidemiological studies of low-dose ionizing radiation and cancer: Summary bias assessment and meta-analysis. *J Natl Cancer Inst Monogr*, 2020(56):188–200. Erratum in: J Natl Cancer Inst Monogr. 2023 May 4;2023(61):e1. PMID: 32657347; PMCID: PMC8454205.

Daniela Hladik, Claudia Dalke, Christine von Toerne, Stefanie M. Hauck, Omid Azimzadeh, Jos Philipp, Marie-Claire Ung, Helmut Schlattl, Ute Rößler, Jochen Graw, Michael J. Atkinson, and Soile Tapio. 2020. Creb signaling mediates dose-dependent radiation response in the murine hippocampus two years after total body exposure. *Journal of Proteome Research*, 19(1):337–345. PMID: 31657930.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Kanchan Jha, Sourav Karmakar, and Sriparna Saha. 2023. Graph-bert and language model-based framework for protein–protein interaction identification. *Scientific Reports*, 13(1):5663.

K. Ji, Y. Wang, L. Du, et al. 2019. Research progress on the biological effects of low-dose radiation in china. *Dose-Response*, 17(1).

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

M. Jimenez-Sanchez, F. Licitra, B. R. Underwood, and D. C. Rubinsztein. 2017. Huntington's disease: Mechanisms of pathogenesis and therapeutic strategies. *Cold Spring Harbor Perspectives in Medicine*, 7(7):a024240.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Rhea Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. 2021. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589. Epub 2021 Jul 15. PMID: 34265844; PMCID: PMC8371605.

Kenji Kamiya, Kotaro Ozasa, Suminori Akiba, Ohstura Niwa, Kazunori Kodama, Noboru Takamura, Elena K Zaharieva, Yuko Kimura, and Richard Wakeford. 2015. Long-term effects of radiation exposure on health. *The Lancet*, 386:469–478. From Hiroshima and Nagasaki to Fukushima.

Minoru Kanehisa, Miho Furumichi, Mao Tanabe, Yoko Sato, and Kanae Morishima. 2017. Kegg: new perspectives on genomes, pathways, diseases and drugs. *Nucleic Acids Res*, 45(D1):D353–D361. Epub 2016 Nov 28. PMID: 27899662; PMCID: PMC5210567.

Md Gulam Musawwir Khan and Yi Wang. 2022. Advances in the current understanding of how low-dose radiation affects the cell cycle. *Cells*, 11(3).

A. Kunchok, A. Zekeridou, and A. McKeon. 2019. Autoimmune glial fibrillary acidic protein astrocytopathy. *Current Opinion in Neurology*, 32(3):452–458.

Y Liu, X Hu, C Han, L Wang, X Zhang, X He, and X Lu. 2015. Targeting tumor suppressor genes for cancer therapy. *Bioessays*, 37:1277–1286.

Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. 2022. Biogpt: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6).

Liuzhenghao Lv, Zongying Lin, Hao Li, Yuyang Liu, Jiaxi Cui, Calvin Yu-Chian Chen, Li Yuan, and Yonghong Tian. 2024. Prollama: A protein large language model for multi-task protein language processing. *Preprint*, arXiv:2402.16445.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and B Bossan. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods. *Younes Belkada and Sayak Paul," PEFT: State-of-the-art Parameter-Efficient Fine-Tuning methods*.

R. Medeiros, D. Baglietto-Vargas, and F. M. LaFerla. 2011. The role of tau in alzheimer's disease and related disorders. *CNS Neurosci Ther*, 17(5):514–524.

MS Mehta, SC Dolfi, R Bronfenbrener, E Bilal, C Chen, D Moore, Y Lin, H Rahim, S Aisner, RD Kersellius, J Teh, S Chen, DL Toppmeyer, DJ Medina, S Ganesan, A Vazquez, and KM Hirshfield. 2013. Metabotropic glutamate receptor 1 expression and its polymorphic variants associate with breast cancer phenotypes. *PLoS One*, 8:e69851.

Erik Nijkamp, Jeffrey Ruffolo, Eli N. Weinstein, Nikhil Naik, and Ali Madani. 2022. Progen2: Exploring the boundaries of protein language models. *Preprint*, arXiv:2206.13517.

Karolin H. Nord, Henrik Lilljebjörn, Francesco Vezzi, Jenny Nilsson, Linda Magnusson, Johnbosco Tayebwa, Danielle de Jong, Judith V. M. G. Bovée, Pancras C. W. Hogendoorn, and Karoly Szuhai. 2014. Grm1 is upregulated through gene fusion and promoter swapping in chondromyxoid fibroma. *Nature Genetics*, 46:474–477.

Rose Oughtred, Jennifer Rust, Christie Chang, Bobby-Joe Breitkreutz, Chris Stark, Andrew Willems, Lorrie Boucher, Genie Leung, Nadine Kolas, Frederick Zhang, Sonam Dolma, Jasmin Coulombe-Huntington, Andrew Chatr-aryamontri, Kara Dolinski, and Mike Tyers. 2021. The biogrid database: A comprehensive biomedical resource of curated protein, genetic, and chemical interactions. *Protein Science*, 30(1):187–200.

Fen Pei, Qingya Shi, Haotian Zhang, and Ivet Bahar. 2021. Predicting protein–protein interactions using symmetric logistic matrix factorization. *Journal of Chemical Information and Modeling*, 61(4):1670–1682. PMID: 33831302.

Jiajun Qiu, Kui Chen, Chunlong Zhong, Sihao Zhu, and Xiao Ma. 2021. Network-based protein-protein interaction prediction method maps perturbations of cancer interactome. *PLOS Genetics*, 17(11):1–19.

Z Schmal, A Isermann, D Hladik, C von Toerne, S Tapio, and CE Rübe. 2019. Dna damage accumulation during fractionated low-dose radiation compromises hippocampal neurogenesis. *Radiotherapy and Oncology*, 137:45–54.

D J Shah, R K Sachs, and D J Wilson. 2014. Radiation-induced cancer: a modern view. *British Journal of Radiology*, 85(1020):e1166–e1173.

Neel K. Sharma, Rupali Sharma, Deepali Mathur, Shashwat Sharad, Gillipsie Minhas, Kulsajan Bhatia, Akshay Anand, and Sanchita P. Ghosh. 2018. Role of ionizing radiation in neurodegenerative diseases. *Frontiers in Aging Neuroscience*, 10.

D. Szklarczyk, A. L. Gable, K. C. Nastou, D. Lyon, R. Kirsch, S. Pyysalo, N. T. Doncheva, M. Legeay, T. Fang, P. Bork, L. J. Jensen, and C. von Mering. 2021. The string database in 2021: customizable protein-protein networks, and functional characterization of user-uploaded gene/measurement sets. *Nucleic Acids Res*, 49(D1):D605–D612. Erratum in: Nucleic Acids Res. 2021 Oct 11;49(18):10800. PMID: 33237311; PMCID: PMC7779004.

Sarah J. Tabrizi, Blair R. Leavitt, G. Bernhard Landwehrmeyer, Edward J. Wild, Carsten Saft, Roger A. Barker, Nick F. Blair, David Craufurd, Josef Priller, Hugh Rickards, Anne Rosser, Holly B. Kordasiewicz, Christian Czech, Eric E. Swayze, Daniel A. Norris, Tiffany Baumann, Irene Gerlach, Scott A. Schobel, Erika Paz, Anne V. Smith, C. Frank Bennett, and Roger M. Lane. 2019. Targeting huntingtin expression in patients with huntington's disease. *New England Journal of Medicine*, 380(24):2307–2316.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. GitHub. Available: `https://crfm.stanford.edu/2023/03/13/alpaca.html`.

Jelena Popović Tatjana Paunesku, Aleksandra Stevanović and Gayle E. Woloschak. 2021. Effects of low dose and low dose rate low linear energy transfer radiation on animals – review of recent studies relevant for carcinogenesis. *International Journal of Radiation Biology*, 97(6):757–768. PMID: 33289582.

M Tessema, CM Yingling, MA Picchi, G Wu, T Ryba, Y Lin, AO Bungum, ES Edell, A Spira, and SA Belinsky. 2017. Ank1 methylation regulates expression of microrna-486-5p and discriminates lung tumors by histology and smoking status. *Cancer Letters*, 410:191–200.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Jin-song Wang, Hai-juan Wang, and Hai-li Qian. 2018. Biological effects of radiation on cancer cells. *Military medical research*, 5:1–10.

Q. Wang, B. Chang, X. Li, and Z. Zou. 2021. Role of aldh2 in hepatic disorders: Gene polymorphism and disease pathogenesis. *J Clin Transl Hepatol*, 9(1):90–98.

Yanbin Wang, Zhu-Hong You, Shan Yang, Xiao Li, Tong-Hai Jiang, and Xi Zhou. 2019. A high efficient biological language model for predicting protein–protein interactions. *Cells*, 8(2).

Xiao X, Liu H, Liu X, Zhang W, Zhang S, and Jiao B. 2021. App, psen1, and psen2 variants in alzheimer's disease: Systematic re-evaluation according to acmg guidelines. *Frontiers in Aging Neuroscience*, 13:695808.

Z. Yang and K. K. Wang. 2015. Glial fibrillary acidic protein: from intermediate filament assembly and gliosis to neurobiomarker. *Trends Neurosci*, 38(6):364–374.

Qiang Zhang, Keyang Ding, Tianwen Lyv, Xinda Wang, Qingyu Yin, Yiwen Zhang, Jing Yu, Yuhao Wang, Xiaotong Li, Zhuoyi Xiang, Xiang Zhuang, Zeyuan Wang, Ming Qin, Mengyao Zhang, Jinlu Zhang, Jiyu Cui, Renjun Xu, Hongyang Chen, Xiaohui Fan, Huabin Xing, and Huajun Chen. 2024. Scientific large language models: A survey on biological & chemical domains. *Preprint*, arXiv:2401.14656.

D. Y. Zhuang, S. X. Ding, F. Wang, X. C. Yang, X. L. Pan, Y. W. Bao, L. M. Zhou, and H. B. Li. 2022. Identification of six novel variants of acad8 in isobutyryl-coa dehydrogenase deficiency with increased c4 carnitine using tandem mass spectrometry and ngs sequencing. *Front Genet*, 12:791869.

A Dataset Information

Dataset 1

The first dataset was provided from this study: "CREB Signaling Mediates Dose-Dependent Radiation Response in the Murine Hippocampus Two Years after Total Body Exposure" (Hladik et al., 2020). This study records the modulation of protein expressions in response to varied radiation exposures, categorizing proteins based on their upregulation or downregulation across three distinct radiation groups. A graphical representation of the significantly deregulated proteins can be seen in Chart A, this chart was presented in the original study and helps to visualize the structure and size of the dataset.

To construct a balanced representation of the data, we combine the identified upregulated and downregulated proteins, for each of the three radiation groups. Subsequently, we employ a randomized selection process, drawing an equitable count of proteins from the list of proteins deemed unaffected by LDR, as shown by the original study.

After cleaning the data, the number of proteins in each of the three subsets (1.1, 1.2, and 1.3) are 892, 1332, and 204 proteins respectively. Each subset maintains an equal distribution between proteins influenced by LDR and those unaffected, thus ensuring analytical balance. The LLMs are then tasked to evaluate the following query for each protein: "Given the options yes or no, will there be significant deregulation of the protein {protein x} 24 months post low-dose radiation exposure at {dosage level} Gy?".

Dataset 2

The second dataset was provided from the study titled "DNA damage accumulation during fractionated low-dose radiation compromises hippocampal Neurogenesis" (Schmal et al., 2019). This research provides an evaluation of protein expression changes due to low-dose radiation (LDR), and gives information regarding the temporal aspects of radiation exposure on cellular processes. Similar to Dataset 1, we have provided

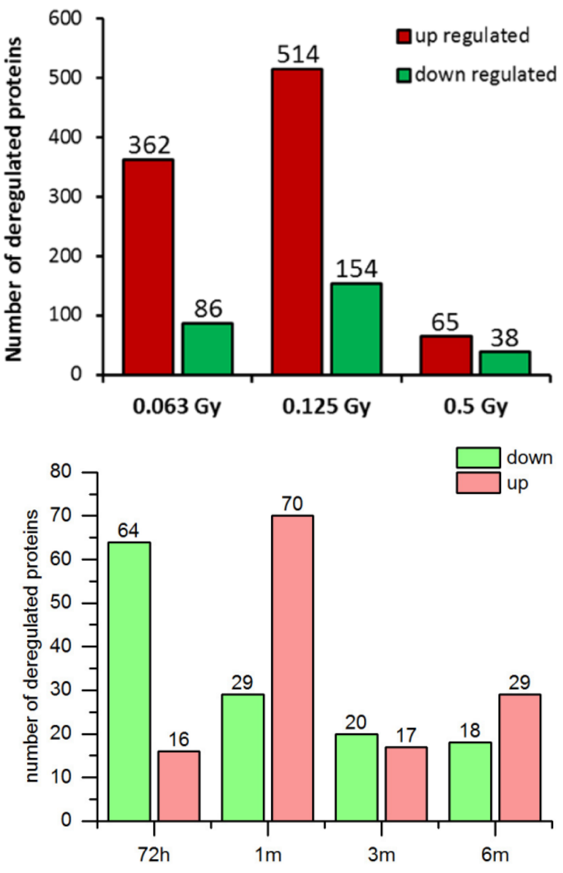


Figure 4: Chart A (Top) and Chart B (Bottom)

a graphical representation of the significantly deregulated proteins in Chart B, this chart was also presented in the original study.

This dataset encapsulates the regulatory status of proteins, upregulated or downregulated, across four distinct cohorts. Each cohort underwent an identical radiation dosage of 2.0 Gy, but the resultant protein expression was analyzed at different post-exposure intervals. Mirroring the methodology applied to the first dataset, we combined the upregulated and downregulated protein expressions, as indicated by the red and green columns for each group, and randomly sample the unaffected proteins. This approach ensures a balanced representation of the data for each group.

After the data cleaning process the number of proteins in each of the four subsets (2.1, 2.2, 2.3, and 2.4) are 160, 198, 74, and 94 proteins respectively. Similar to the first dataset, the LLMs are then tasked to evaluate the following query for each protein: "Given the options yes or no, will there be significant deregulation of the protein {protein x} {time} after exposure to low dose radiation at 2.0 Gy?".

Dataset 3

Dataset 3 was provided from the study titled "Low-dose radiation differentially regulates protein acetylation and histone deacetylase expression in human coronary artery endothelial cells" (Barjaktarovic et al., 2017). This work delves into the post-translational modifications, specifically acetylation, that occur in proteins of human coronary artery endothelial cells as a result of low-dose radiation (LDR) exposure. The administered radiation dose of 0.5 Gy and the subsequent temporal protein measurements offer valuable insights into the cellular responses.

In this study, the protein deregulation via acetylation was monitored at two time intervals: at 4 hours and then at 24 hours post-radiation exposure. The resulting subsets for analysis, capturing the 4 hour period and the 24 hour period, comprised 98 and 154 proteins, respectively. These two groups represents datasets 3.1 and 3.2.

To maintain a consistent evaluation strategy, the LLMs are given a prompt for each protein in the dataset: "Given the options yes or no, will there be an altered acetylation status of protein {protein x} 24 hours after exposure to low dose radiation at 0.5 Gy?".

Dataset 3c

Dataset 3c represents a strategic combination of datasets 1, 2, and 3. This integration was motivated by insights derived from the review of experiments 1 through 3, which suggested limitations in the approach's efficacy. Specifically, the chosen prompts for these experiments were potentially too narrowly defined, and the datasets themselves were not sufficiently sized to enable the LLMs to recognize the patterns within the data.

To address these challenges, we synthesized a comprehensive dataset combining the protein deregulation data from the first 3 datasets. The objective was to refine the training process for the LLMs using a larger dataset and significantly broader prompting strategy. The reformulated prompt used to train the LLMs is: "Given the options yes or no, will there be deregulation of the protein {protein x} after low-dose radiation exposure?".

Dataset 3c includes an amalgamation of datasets 1.1, 1.2, 1.3, 2.1, 2.2, 2.3, 2.4, 3.1, 3.2. It is a combination of the proteins in each of the columns from charts A and B from Figure 17, along with the proteins from datasets 3.1 and 3.2. The repeated proteins between all datasets were removed. These proteins become deregulated across different time intervals and radiation dosage levels, resulting in a comprehensive dataset of 1,111 proteins.

A randomized sampling methodology was employed to select proteins that do not exhibit deregulation across these varied experimental conditions, which resulted in a dataset featuring 2,222 proteins. This augmented dataset size significantly enhances the LLMs' training ability, facilitating a more nuanced understanding of protein behavior in response to low-dose radiation exposure.

Dataset 4

Dataset 4 was provided by the study "Predicting Protein-Protein Interactions Using Symmetric Logistic Matrix Factorization" (Pei et al., 2021). In an effort to understand the Protein-Protein Interactions (PPIs) specific to disease mechanisms, this dataset focuses on the protein interactions associated with neurodegenerative diseases.

From the data provided, this study narrows its focus to a subset encompassing 820 proteins, which form a network of 5,881 positive (interaction present) and 5,881 negative (interaction absent) protein pairs. This gives us a total of 11,762 protein interactions. The LLMs are prompted with the following query for each protein pair: "Given the options yes or no, do proteins {protein x} and {protein y} interact in the presence of neurodegenerative disease?".

Dataset 5

Concurrent with the exploration of neurodegenerative diseases in Dataset 4, Dataset 5 focuses on metabolic disorders. Provided by the same study "Predicting Protein-Protein Interactions Using Symmetric Logistic Matrix Factorization" (Pei et al., 2021), this dataset shines a light on the protein interactions that might contribute to metabolic dysfunction.

This data is made up of 1,063 proteins, from which a balanced collection of 5,131 positive and 5,131 negative protein pairs is drawn. This leads to a total dataset size of 10,262 protein interactions. The LLMs will use a similar prompt to that used for dataset 4: "Given the options yes or no, do proteins {protein x} and {protein y} interact in the presence of a metabolic disorder?".

Dataset 6

Dataset 6 was provided from the study "Network-based protein-protein interaction prediction method maps perturbations of cancer interactome" (Qiu et al., 2021), which offers a focused lens on the protein interaction network within the context of cancer.

This data presents a network of protein interactions consisting of 933 positive instances—indicative of an interaction's presence—and 1,308 negative instances, signifying the absence of interaction. To achieve an even representation akin to previous datasets, we conduct a randomized selection, reducing the negative instances to 933, thereby equalizing the number of positive and negative samples and giving a total of 1,866 protein interactions. The prompt used for this dataset is similar to datasets 4 and 5: "Given the options yes or no, do proteins {protein x} and {protein y} interact in the presence of cancer?".

XrayGPT: Chest Radiographs Summarization using Large Medical Vision-Language Models

Omkar Thawakar[1] Abdelrahman Shaker[1] Sahal Shaji Mullappilly[1] Hisham Cholakkal[1]
Rao Muhammad Anwer[1,2] Salman Khan[1] Jorma Laaksonen[2] Fahad Shahbaz Khan[1]
[1]Mohamed bin Zayed University of AI [2]Aalto University

Abstract

The latest breakthroughs in large language models (LLMs) and vision-language models (VLMs) have showcased promising capabilities toward performing a wide range of tasks. Such models are typically trained on massive datasets comprising billions of image-text pairs with diverse tasks. However, their performance on task-specific domains, such as radiology, is still under-explored. While few works have recently explored LLMs-based conversational medical models, they mainly focus on text-based analysis. In this paper, we introduce XrayGPT, a conversational medical vision-language (VLMs) model that can analyze and answer open-ended questions about chest radiographs. Specifically, we align both medical visual encoder with a fine-tuned LLM to possess visual conversation abilities, grounded in an understanding of radiographs and medical knowledge. For improved alignment of chest radiograph data, we generate 217k interactive and high-quality summaries from free-text radiology reports. Extensive experiments are conducted to validate the merits of XrayGPT. To conduct an expert evaluation, certified medical doctors evaluated the output of our XrayGPT on a test subset and the results reveal that more than 70% of the responses are scientifically accurate, with an average score of 4/5. Our code and models are available at: https://github.com/mbzuai-oryx/XrayGPT

1 Introduction

The Large-scale Vision-Language models have emerged as a transformative area of research at the intersection of computer vision and natural language processing, enabling machines to understand and generate information from both visual and textual modalities. These models represent a significant advancement in the field, bridging the gap between visual perception and language comprehension, and have demonstrated remarkable capabilities across various tasks, including but not limited to image captioning (Hossain et al., 2019), visual question answering (Lu et al., 2023), and visual commonsense reasoning (Zellers et al., 2019). Training these models requires vast amounts of image and text data, enabling them to learn rich representations that capture the intricacies of both modalities. Additionally, fine-tuning can be employed using task-specific data to better align the models with specific end tasks and user preferences. Recently, Bard and GPT-4 have demonstrated impressive capabilities in various tasks, raising excitement within the research community and industry. However, it is important to note that the models of Bard and GPT-4 are not currently available as open-source, limiting access to their underlying architecture and implementation details.

Recently, Mini-GPT (Zhu et al., 2023) demonstrated a range of impressive capabilities by aligning both vision and language models. It excels at generating contextual descriptions based on the given image. However, it is not as effective in medical scenarios due to the significant differences between medical image-text pairs and general web content. Adopting vision-text pre-training in the medical domain is a challenging task because of two factors: (1) Lack of data, Mini-GPT has trained the projection layer on a dataset of 5M image-text pairs, while the total number of publicly available medical images and reports is orders of magnitude below. (2) Different modalities and domains, while Mini-GPT may involve distinguishing between broad categories like "Person" and "Car" the distinctions within medical domains are much more subtle and fine-grained. For instance, differentiating between terms like "Pneumonia" and "Pleural Effusion" requires more precision by capturing and aligning relevant medical domain knowledge.

Chest radiographs are vital for clinical decision-making as they offer essential diagnostic and prognostic insights about the patients. Text summarization tasks can partially address this challenge by

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 440–448
August 16, 2024. ©2024 Association for Computational Linguistics

providing meaningful information and summaries based on the given radiology reports. In our approach, we go beyond traditional summarization techniques by providing concise summaries that highlight the key findings and the impression based on the X-ray. Additionally, our model allows for interactive engagement, enabling users to ask follow-up questions based on the provided answers. We argue that based on the visual and large language models, the majority of knowledge acquired during the pertaining stage of these models requires a domain-specific high-quality instruction set derived from task-specific data to achieve promising results. The main contributions of our work are:-

- We generate interactive and clean summaries (217k) from free-text radiology reports of two datasets: MIMIC-CXR (Johnson et al., 2019) and OpenI (Demner-Fushman et al., 2015). These summaries serve to enhance the performance of our XrayGPT by fine-tuning the linear transformation layer on high-quality data.
- We fine-tune a standard LLM (Vicuna) on medical data (100k real conversations) and 20k radiology conversations from samples from chatdoctor (Li et al., 2023) to acquire medical domain-specific features.
- In our XrayGPT, the frozen specialized medical visual encoder is aligned with a fine-tuned medical LLM, using a simple linear transformation to understand medical meanings and acquire visual conversation capabilities.
- We conduct experiments including an evaluation study through certified medical doctors to validate the merits of our XrayGPT. To promote future research, our codebase, fine-tuned models, and high-quality instruction set along with the recipe for data generation and model training will be publicly released.

2 Related Work

Medical Chatbots: Recent medical chatbots have emerged as valuable tools in healthcare, providing personalized support and assistance to patients and professionals. The recently introduced Chatdoctor (Li et al., 2023), a next-gen AI doctor based on LLaMA (Touvron et al., 2023), aims to be an intelligent healthcare companion, answering patient queries and offering personalized medical advice. After success of ChatGPT (OpenAI, 2022), GPT-4 (OpenAI, 2023) and other open source LLM's (Touvron et al., 2023; Chiang et al., 2023; Taori et al., 2023), many medical chatbots were introduced recently such as Med-Alpaca (Han et al., 2023), PMC-LLaMA (Wu et al., 2023), and DoctorGLM (Xiong et al., 2023). These models utilize open-source LLMs and are finetuned on medical instructions, demonstrating the potential of chatbots to enhance patient engagement and health outcomes with personalized interactions.

Large Vision-Language Models: A significant area of research in natural language processing (NLP) and computer vision is the exploration of Large Vision-Language Model (VLM) learning techniques. This VLM aims to bridge the gap between visual and textual information, enabling machines to understand and generate content that combines both modalities. Recent studies have demonstrated the potential of VLM models in various tasks, such as image captioning (Zhu et al., 2023), visual question answering (Bazi et al., 2023; Liu et al., 2023; Muhammad Maaz and Khan, 2023), and image generation (Zhang and Agrawala, 2023).

3 Method

XrayGPT is an innovative conversational medical vision-language model designed for analyzing chest radiographs. Our approach draws inspiration from the design of vision-language models in general, but with a specific focus on the medical domain. Due to the limited availability of medical image-summary pairs, we adopt a similar methodology by building upon a pre-trained medical vision encoder (VLM) and medical large language model (LLM), as our foundation. The fine-tuning process involves aligning both modalities using high-quality image-summary pairs through a simple transformation layer. This alignment enables XrayGPT to possess the capability of generating insightful summaries about chest radiographs.

3.1 Model Architecture

We show in Fig. 1 an overview of our XrayGPT. Given the X-ray, we align both visual features and textual information from a pre-trained medical vision encoder (VLM), and medical large language model (LLM). Specifically, we utilize MedClip (Wang et al., 2022) as a visual encoder and our large language model (LLM) is built upon the recent Vicuna (Chiang et al., 2023).

Given X-ray $\mathbf{x} \in R^{H \times W \times C}$, the visual encoder is used to encode the image into embeddings using

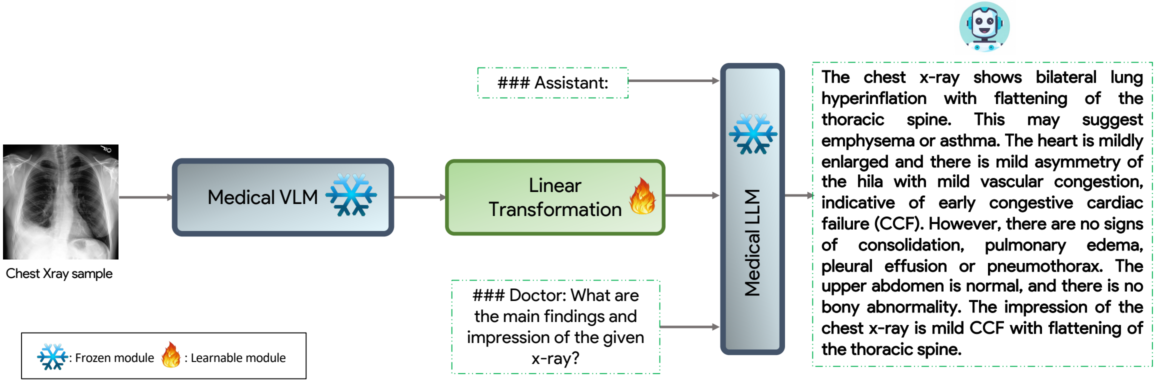

### Assistant:
Chest Xray sample
Medical VLM
Linear Transformation
Medical LLM
### Doctor: What are the main findings and impression of the given x-ray?
The chest x-ray shows bilateral lung hyperinflation with flattening of the thoracic spine. This may suggest emphysema or asthma. The heart is mildly enlarged and there is mild asymmetry of the hila with mild vascular congestion, indicative of early congestive cardiac failure (CCF). However, there are no signs of consolidation, pulmonary edema, pleural effusion or pneumothorax. The upper abdomen is normal, and there is no bony abnormality. The impression of the chest x-ray is mild CCF with flattening of the thoracic spine.
: Frozen module
: Learnable module

Figure 1: Overview of our XrayGPT framework. The input X-ray is passed sequentially to three components. (i) Frozen medical visual encoder to extract relevant features pertaining to the chest diagnosis. (ii) Learnable linear transformation layer to align the medical visual features with the Medical LLM to learn medical visual-text alignment. (iii) Frozen Medical LLM to generate X-ray summary based on encoded features and the given prompt.

a vision encoder E_{img}. Then, the raw embeddings are mapped to an output dimension of 512 using a linear projection head f_v.

$$\mathbf{V}_p = f_v(E_{img}(\mathbf{x})) \tag{1}$$

To bridge the gap between image-level features and the language decoder's embedding space, we employ a trainable linear transformation layer, denoted as t. This layer projects the image-level features, represented by $\mathbf{V}_p$, into corresponding language embedding tokens, denoted as $\mathbf{L}_v$:

$$\mathbf{L}_v = t(v_p), \tag{2}$$

We use pre-defined prompts as directives for our LLM in the given format ###Doctor: <Img><ImgFeature></Img> <Instruction> ###Assistant, where ###Doctor, corresponds to the prompt. ###Assistant, serves the purpose of determining the system role, which in our case is defined as "*You are a helpful healthcare virtual assistant.*" <Instruction> refers to a randomly selected instruction from our pre-defined set. Both text queries undergo tokenization, resulting in dimensions represented by $\mathbf{L}_t$. Finally, $\mathbf{L}_v$ is concatenated with $\mathbf{L}_t$ and fed into the medical LLM, fine-tuned Vicuna, which generates the summary of the chest x-ray.

Our XrayGPT follows a two-stage training approach. In the first stage, we pre-train the model using high-quality curated interactive summaries of the training set of MIMIC-CXR (Johnson et al., 2019) reports. While in the second stage, we use the curated interactive summaries of OpenI (Demner-Fushman et al., 2015) reports.

3.2 Image-text alignment

To align the generated high-quality summaries with the given x-ray, we use similar conversational format of the Vicuna (Chiang et al., 2023) language model as follows:

###Doctor: $X_R X_Q$ *###Assistant:* X_S

where X_R is the visual representation produced by the linear transformation layer for image X, X_Q is a sampled question (e.g. What are the main findings and impression of the given X-ray?), and X_S is the associated summary for image X.

4 Curating high-quality data

Datasets: The MIMIC-CXR consists of a collection of chest radiographs associated with free-text radiology reports. It consists of 377,110 images and 227,827 associated reports, which are used for both training and testing purposes. The dataset is de-identified by removing the health information to satisfy health insurance and privacy requirements. The OpenI dataset is a collection of chest X-ray images from the Indiana University hospital network, composing 6,459 images and 3,955 reports.

High Quality and Interactive Summaries: To generate concise and coherent medical summaries from the unstructured reports, we perform the following pre-processing steps for both datasets: (1) Removal of incomplete reports lacking finding or impression sections. (2) Elimination of reports that have finding sections containing less than 10 words. (3) Exclusion of reports with impression sections containing less than 2 words.

In addition, utilizing the power of gpt-3.5-turbo model, we further implement the following pre-processing techniques to ensure high-quality summaries per image: (1) Elimination of sentences containing comparisons to the patient's prior medical history. (2) Removal of de-defined symbols "__", while preserving the original meaning. (3) As our training is based on image-text pairs, we excluded the provided view from the summary. (4) We combine the clean findings and impressions to generate an interactive and high-quality summary.

Following these steps, we obtained a set of filtered training reports consisting of 114,690 reports associated with 241k training images based on Mimic-CXR dataset. Also, we obtain 3,403 high-quality summaries that used for training based on the OpenI dataset. We show an example before and after our pre-processing in Appendix A.2.

5 Experiments

5.1 Implementation Details

Stage-1 Training: The model is designed in this stage to gain understanding of how Xray image features and corresponding reports are interconnected by analyzing a large set of image-text pairs. We employ our high-quality interactive report summaries as described in sec. 4 of MIMIC-CXR (Johnson et al., 2019) train set with 213,514 image-text pairs. The model is trained for 320k steps with a total batch size of 128 using 4 AMD MI250X GPUs. **Stage-2 Training:** The pretrained model of stage-1 is fine-tuned on 3k highly curated image-text pairs from OpenI dataset (Demner-Fushman et al., 2015). The total training steps are 5k, with a total batch size of 32 using single AMD MI250X GPU.

5.2 Evaluation Metrics

We use the ROUGE Score as an evaluation metric to assess the contributions of our components over the baseline Mini-GPT (Zhu et al., 2023). The ROUGE Score serves as a valuable quantitative measure to assess the performance of different text generation models with the ground truth. Then, we use GPT-based evaluation schema to assess the quality and coherence of the text generated by our approaches, compared to the baseline. Furthermore, we provide certified medical doctors evaluation for 50 samples derived from the testing set of MIMIC-CXR (Johnson et al., 2019).

5.3 Results

In this section, we highlight a key contribution of our XrayGPT compared to our baseline (Zhu et al., 2023). We conduct quantitative evaluation using advanced metrics such as ROUGE score and GPT-based evaluation as described in sec. 5.2. Tab. 1 shows comparison of our key components when progressively integrated into our baseline (Zhu et al., 2023) frame. From Tab. 1 our XrayGPT (row 4) has a significant improvement of 19% over the state-of-the-art baseline (Zhu et al., 2023) on the MIMIC-CXR test set. Also, we did LLM's based evaluation by asking ChatGPT model to choose "which response is closer to reference between baseline vs XrayGPT" where our model scored 82% compared to baseline 6% showing the superiority of XrayGPT for radiology-specific summary.

Method	R-1	R-2	R-L
Baseline	0.1313	0.0221	0.0879
+ MedCLIP	0.1517	0.0308	0.0973
+ MedVicuna	0.2099	0.0551	0.1284
+ RadVicuna	**0.3213**	**0.0912**	**0.1997**

Table 1: Comparison of our XrayGPT components with the baseline Minigpt-4 (Zhu et al., 2023) using ROUGE scores (R-1, R-2, and R-L) on MIMIC-CXR (Johnson et al., 2019) test set. Our approach outperforms Minigpt-4 with an absolute gain of 19% in terms of R-1 score.

To assess the responses of XrayGPT scientifically, we asked certified medical doctors to evaluate the responses of our model alongside the baseline, compared to their real findings and impression. This evaluation shows that our XrayGPT has accurate output in 72% of the cases, with an average score of 4/5, outperforming the baseline which achieves only 20%, with an average score of 2/5. Both models provided inaccurate responses in 8% of the cases. Despite occasional inaccuracies, XrayGPT significantly improves upon the baseline, highlighting its potential in assisting radiologists with chest radiograph analysis. Additional details of our evaluations are in Appendix (A.3,A.4). We also show qualitative examples in Appendix A.5.

6 Conclusion

To conclude, we introduce XrayGPT, an innovative medical vision-language model that combines both vision-language modalities to summarize and answer inquiries regarding chest radiographs. By aligning both modalities and leveraging our proposed interactive summaries derived from free-text radiology reports, XrayGPT demonstrates exceptional visual conversation abilities grounded in a deep understanding of chest radiographs.

7 Limitations

While our XrayGPT shows promise toward constructing conversational VLMs for chest radiograph summarization, we acknowledge some limitations here. We observe some potential limitations in generation of responses when presented with side views, as the majority of the trained images primarily consist of frontal views. To address this limitation, our potential future research direction is to focus on enhancing the quality and reliability of the model when handling multiple views of X-rays. Our current model is specifically trained and designed to answer questions pertaining to chest radiographs. Expanding the model's support to encompass multiple modalities is a potential research direction. By broadening its capabilities, the resulting potential model is expected to possess certain capabilities across various medical imaging domains beyond chest radiographs.

References

Yakoub Bazi, Mohamad Mahmoud Al Rahhal, Laila Bashmal, and Mansour Zuair. 2023. Vision–language model for visual question answering in medical imagery. *Bioengineering*, 10(3):380.

Wei-Lin Chiang et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Dina Demner-Fushman, Marc D. Kohli, Marc B. Rosenman, Sonya E. Shooshan, Laritza Rodriguez, Sameer Antani, George R. Thoma, and Clement J. McDonald. 2015. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association*, 23(2):304–310.

Tianyu Han et al. 2023. Medalpaca–an open-source collection of medical conversational ai models and training data. *arXiv preprint arXiv:2304.08247*.

MD Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. 2019. A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CsUR)*, 51(6):1–36.

Alistair E. W. Johnson et al. 2019. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific Data*.

Yunxiang Li, Zihan Li, Kai Zhang, Ruilong Dan, Steve Jiang, and You Zhang. 2023. Chatdoctor: A medical chat model fine-tuned on a large language model meta-ai (llama) using medical domain knowledge.

Yunyi Liu, Zhanyu Wang, Dong Xu, and Luping Zhou. 2023. Q2atransformer: Improving medical vqa via an answer querying decoder. *arXiv preprint arXiv:2304.01611*.

Siyu Lu, Yueming Ding, Mingzhe Liu, Zhengtong Yin, Lirong Yin, and Wenfeng Zheng. 2023. Multiscale feature extraction and fusion of image and text in vqa. *International Journal of Computational Intelligence Systems*, 16(1):54.

Salman Khan Muhammad Maaz, Hanoona Rasheed and Fahad Khan. 2023. Video-chatgpt: Towards detailed video understanding via large vision and language models. *ArXiv 2306.05424*.

OpenAI. 2022. Chatgpt.

OpenAI. 2023. Gpt-4 technical report.

Rohan Taori et al. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

Hugo Touvron et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Zifeng Wang, Zhenbang Wu, Dinesh Agarwal, and Jimeng Sun. 2022. Medclip: Contrastive learning from unpaired medical images and text. *arXiv preprint arXiv:2210.10163*.

Chaoyi Wu, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2023. Pmc-llama: Further fine-tuning llama on medical papers. *arXiv preprint arXiv:2304.14454*.

Honglin Xiong et al. 2023. Doctorglm: Fine-tuning your chinese doctor is not a herculean task. *arXiv preprint arXiv:2304.01097*.

Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6720–6731.

Lvmin Zhang and Maneesh Agrawala. 2023. Adding conditional control to text-to-image diffusion models.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

A Appendix

A.1 Medical Data

We gather the 100k medical conversational samples data and 20k radiology specific samples from chatdoctor (Li et al., 2023) and train LLM (vicuna) to obtain medical domain-specific LLM. This medical specific LLM then helps in better alignment of linear transformation layer of our XrayGPT. These medical conversations are between patient queries and a doctor response to it. On average a question length is about 45 words, and a doctor response covers a brief answer comprising of 100 words. Below is an example of such a conversation sample. These conversations cover a variety of topics such as radiology, paleontology, histopathology, etc.

A.2 Example for generating interactive summary

Input findings: *PA and lateral views of the chest were provided demonstrating no focal consolidation, effusion, or pneumothorax. Cardiomediastinal silhouette appears normal and stable. There is a compression deformity involving a mid thoracic vertebral body, which appears new from the prior chest radiograph of ___. No free air below the right hemidiaphragm. There are tiny surgical clips in the left base of neck, likely indicating prior thyroid surgery.* **Input Impression**: *No acute intrathoracic process. Interval development of a mid thoracic spine compression fracture.* **High-quality and interactive summary**: *The chest x-ray findings reveal no evidence of focal consolidation, effusion, or pneumothorax. The cardiomediastinal silhouette appears stable and normal. There is a newly developed mid thoracic spine compression fracture but no free air below the right hemidiaphragm. The presence of surgical clips in the left base of the neck suggests prior thyroid surgery. The impression suggests that there is no acute intrathoracic condition detected in the x-ray aside from the new development of mid thoracic spine compression fracture.*

A.3 LLM based Evaluation

LLM-based evaluation represents a comprehensive and meticulous approach to assessing and analyzing language models such as GPT-3.5 (OpenAI, 2022, 2023). This evaluation methodology aims to evaluate the language model's performance, capabilities, and constraints when generating text that is both coherent and contextually appropriate. The evaluation process employs a diverse range of techniques, including automated metrics like perplexity and fluency scores, as well as human evaluations that rely on expert judgments and comparisons with reference texts. By incorporating quantitative and qualitative measures, LLM-based evaluation offers valuable insights into the language model's grasp of language comprehension, coherence, factual accuracy, and its ability to produce responses that align with the given context. Through this robust evaluation framework, we can continuously enhance and refine language models by addressing potential biases, improving response quality, and maximizing their practicality across various language tasks and domains. We used GPT-3.5 (OpenAI, 2022) Turbo for evaluation of our baseline (Zhu et al., 2023) vs XrayGPT generated responses with the following meta details.

System Role: You are a chest radiologist that evaluates the response of two models: Model_1 and Model_2 and say which one is closer to the ground truth. You should print which model is closer to the ground truth.

User Role: Perform the following task: **[1]** Which model (Model_1 or Model_2) is closer to the Ground_Truth based on the medical finding and impression? Ground_Truth: <response> Model_1: <response> Model_2: <response>. **[2]** Output should be in valid JSON format without explanation where key=answer and value should be Model_1 or Model_2.

A.4 Evaluation from Certified Medical Doctors

In order to comprehensively evaluate the quality and performance of XrayGPT-generated samples, we conducted an extensive assessment in collaboration with certified medical doctors, utilizing samples from MIMIC-CXR test split. During the evaluation process, we carefully curated summaries generated by two different methods: our baseline (Zhu et al., 2023) method and the XrayGPT model. These summaries were then presented to the certified medical doctors, who were tasked with determining which response provided a relatively superior summary and diagnosis for the given X-ray image. To further gauge the medical doctor's evaluation, we requested them to assign a score ranging from 1 to 5 to each response, indicating the perceived quality of the summaries. This detailed evaluation process allowed us to gather valuable insights into the comparative performance and effectiveness of the baseline (Zhu et al., 2023) method and the XrayGPT model in generating accurate and informative summaries for X-ray images.

Our scoring criteria is defined as follows,

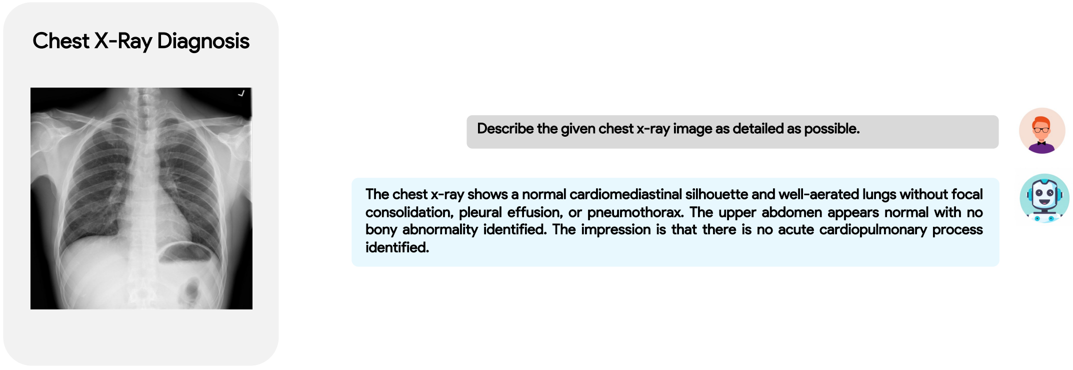
1. poor description
2. fair description
3. moderate description
4. good description
5. excellent description

In this context, an 'excellent description' is defined as one that accurately reflects the findings and impression of the X-ray without any extraneous false positives or negatives. A 'good description', meanwhile, includes all necessary findings and impressions but may contain additional false positive findings that do not detract from the overall impression. XrayGPT has an accurate output in 72% of the cases. That means in the evaluation of XrayGPT responses by medical professionals, our XrayGPT has a score of 4 or higher in 72% of the responses.

During the evaluation process, medical doctors also identify and address limitations and challenges associated with our XrayGPT-generated X-ray image summaries. They consider scenarios where our XrayGPT method may fail to detect subtle abnormalities, mischaracterize findings, or provide inaccurate recommendations. The medical doctors explore the causes of such limitations, such as dataset biases, rare or unusual cases not adequately represented during training, or technical factors that may impact the XrayGPT system's performance.

A.5 Qualitative examples

We present a series of illustrative examples that highlight the remarkable capabilities of our XrayGPT. In Fig. 2, our XrayGPT effectively summarizes the contents of a provided chest x-ray image, extracting essential findings and generating an impression. While in Fig. 3, the model demonstrates its proficiency in follow-up procedures by recommending appropriate treatments based on the given case. In Fig. 4, the model accurately identifies key findings in the x-ray and provides correct responses to user queries. Notably, this particular case reveals the presence of lung hyperinflation. Fig. 5 serves as a testament to the versatility of XrayGPT, showcasing its ability to address multiple subsequent questions and suggest treatment options for pleural effusion. Lastly, in Fig. 6, the model showcases its competence in identifying various abnormalities within a chest x-ray while also offering medical treatment recommendations for pulmonary edema. Through these diverse examples, we demonstrate how XrayGPT consistently delivers accurate and comprehensive analyses, positioning it as a valuable tool in the field of medical imaging.

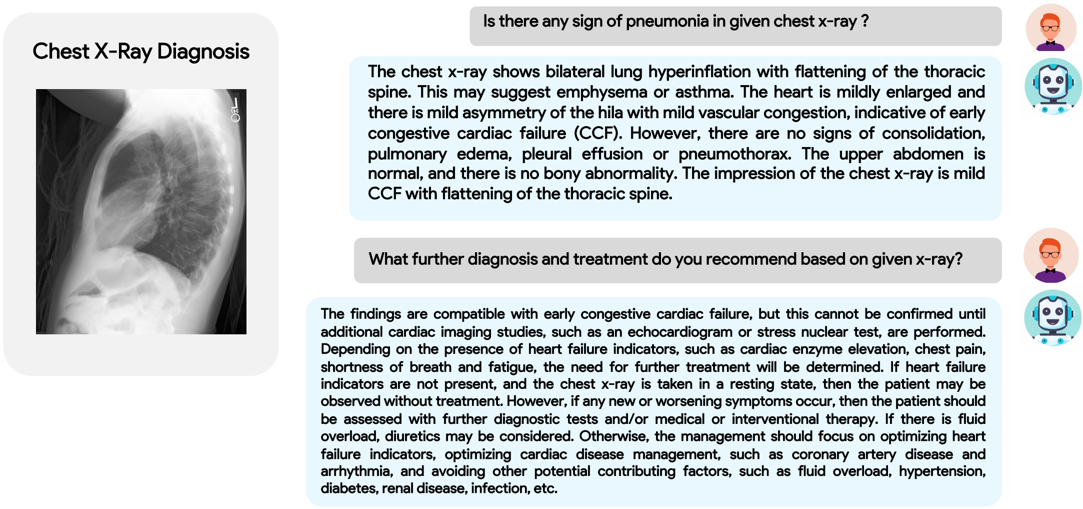


Chest X-Ray Diagnosis

Describe the given chest x-ray image as detailed as possible.

The chest x-ray shows a normal cardiomediastinal silhouette and well-aerated lungs without focal consolidation, pleural effusion, or pneumothorax. The upper abdomen appears normal with no bony abnormality identified. The impression is that there is no acute cardiopulmonary process identified.

Figure 2: **XrayGPT: Delivering Comprehensive and Succinct Findings and Impressions for Every X-ray Image.** Our XrayGPT provides detailed radiology specific summary as well as concise findings and impression for given xray image.



Chest X-Ray Diagnosis

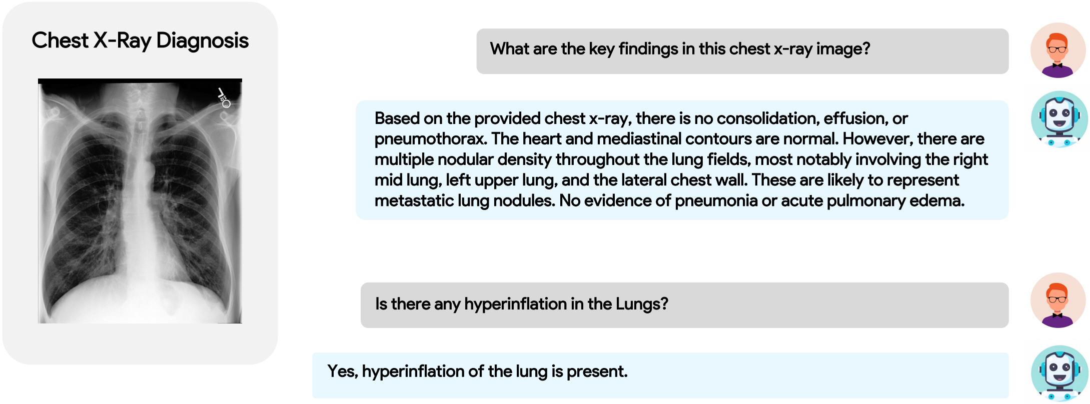
Is there any sign of pneumonia in given chest x-ray ?

The chest x-ray shows bilateral lung hyperinflation with flattening of the thoracic spine. This may suggest emphysema or asthma. The heart is mildly enlarged and there is mild asymmetry of the hila with mild vascular congestion, indicative of early congestive cardiac failure (CCF). However, there are no signs of consolidation, pulmonary edema, pleural effusion or pneumothorax. The upper abdomen is normal, and there is no bony abnormality. The impression of the chest x-ray is mild CCF with flattening of the thoracic spine.

What further diagnosis and treatment do you recommend based on given x-ray?

The findings are compatible with early congestive cardiac failure, but this cannot be confirmed until additional cardiac imaging studies, such as an echocardiogram or stress nuclear test, are performed. Depending on the presence of heart failure indicators, such as cardiac enzyme elevation, chest pain, shortness of breath and fatigue, the need for further treatment will be determined. If heart failure indicators are not present, and the chest x-ray is taken in a resting state, then the patient may be observed without treatment. However, if any new or worsening symptoms occur, then the patient should be assessed with further diagnostic tests and/or medical or interventional therapy. If there is fluid overload, diuretics may be considered. Otherwise, the management should focus on optimizing heart failure indicators, optimizing cardiac disease management, such as coronary artery disease and arrhythmia, and avoiding other potential contributing factors, such as fluid overload, hypertension, diabetes, renal disease, infection, etc.

Figure 3: **XrayGPT: Empowering Users with Concise Insights - Explore the Comprehensive Analysis Derived from the xray Image.** Our XrayGPT provides a brief explanation for users quaries based on the overall findings from the given xray image.



Chest X-Ray Diagnosis

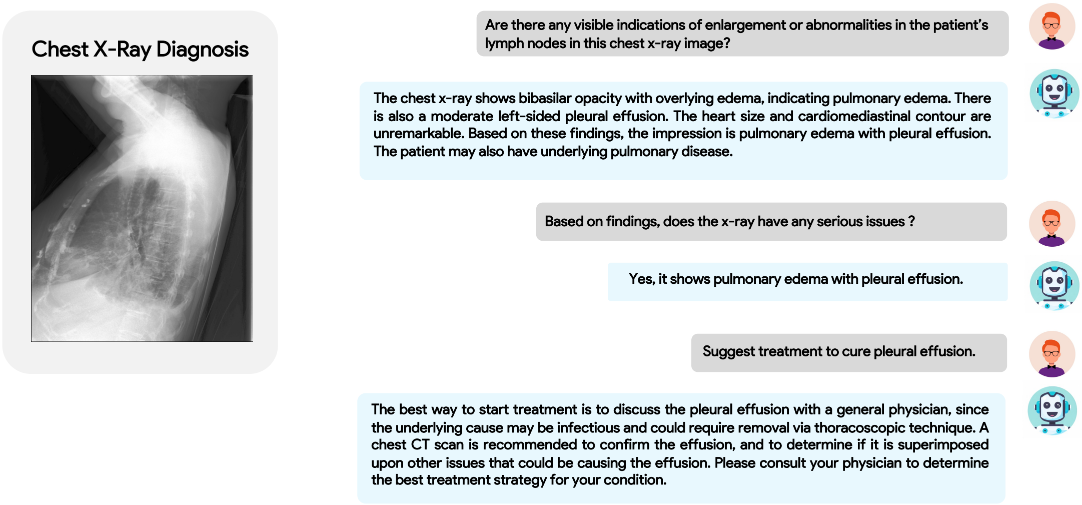
What are the key findings in this chest x-ray image?

Based on the provided chest x-ray, there is no consolidation, effusion, or pneumothorax. The heart and mediastinal contours are normal. However, there are multiple nodular density throughout the lung fields, most notably involving the right mid lung, left upper lung, and the lateral chest wall. These are likely to represent metastatic lung nodules. No evidence of pneumonia or acute pulmonary edema.

Is there any hyperinflation in the Lungs?

Yes, hyperinflation of the lung is present.

Figure 4: **XrayGPT: Unlocking the Power of Precision, Providing Specific Key Details in Response to User Queries.** Our XrayGPT is able to answer specific key detail in response to user queries.



Are there any visible indications of enlargement or abnormalities in the patient's lymph nodes in this chest x-ray image?

The chest x-ray shows bibasilar opacity with overlying edema, indicating pulmonary edema. There is also a moderate left-sided pleural effusion. The heart size and cardiomediastinal contour are unremarkable. Based on these findings, the impression is pulmonary edema with pleural effusion. The patient may also have underlying pulmonary disease.

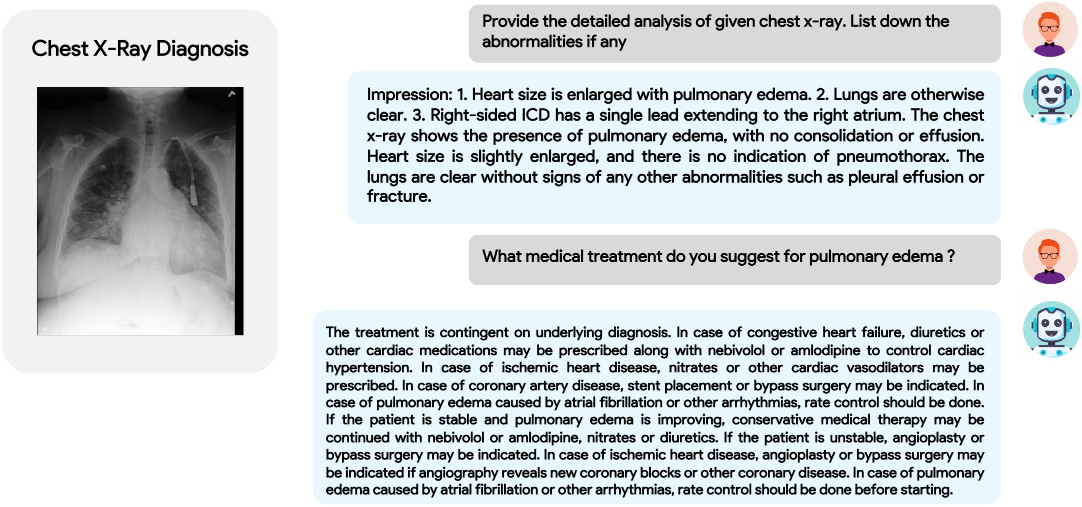
Based on findings, does the x-ray have any serious issues ?

Yes, it shows pulmonary edema with pleural effusion.

Suggest treatment to cure pleural effusion.

The best way to start treatment is to discuss the pleural effusion with a general physician, since the underlying cause may be infectious and could require removal via thoracoscopic technique. A chest CT scan is recommended to confirm the effusion, and to determine if it is superimposed upon other issues that could be causing the effusion. Please consult your physician to determine the best treatment strategy for your condition.

Figure 5: **XrayGPT: The Conversational AI Revolutionizing Radiological Interactions.** Our XrayGPT has radiological conversational capabilities.



Provide the detailed analysis of given chest x-ray. List down the abnormalities if any

Impression: 1. Heart size is enlarged with pulmonary edema. 2. Lungs are otherwise clear. 3. Right-sided ICD has a single lead extending to the right atrium. The chest x-ray shows the presence of pulmonary edema, with no consolidation or effusion. Heart size is slightly enlarged, and there is no indication of pneumothorax. The lungs are clear without signs of any other abnormalities such as pleural effusion or fracture.

What medical treatment do you suggest for pulmonary edema ?

The treatment is contingent on underlying diagnosis. In case of congestive heart failure, diuretics or other cardiac medications may be prescribed along with nebivolol or amlodipine to control cardiac hypertension. In case of ischemic heart disease, nitrates or other cardiac vasodilators may be prescribed. In case of coronary artery disease, stent placement or bypass surgery may be indicated. In case of pulmonary edema caused by atrial fibrillation or other arrhythmias, rate control should be done. If the patient is stable and pulmonary edema is improving, conservative medical therapy may be continued with nebivolol or amlodipine, nitrates or diuretics. If the patient is unstable, angioplasty or bypass surgery may be indicated. In case of ischemic heart disease, angioplasty or bypass surgery may be indicated if angiography reveals new coronary blocks or other coronary disease. In case of pulmonary edema caused by atrial fibrillation or other arrhythmias, rate control should be done before starting.

Figure 6: **XrayGPT: Medical Treatment Recommendation.** Our XrayGPT has the capability to suggest treatment based on the diagnosis.

Multilevel Analysis of Biomedical Domain Adaptation of Llama 2: What Matters the Most? A Case Study

V. Ivan Sanchez Carmona[1] and **Shanshan Jiang**[1] and **Takeshi Suzuki**[2] and **Bin Dong**[1]
[1]Ricoh Software Research Center (Beijing) Co., Ltd
[2]Ricoh
{Vicente.Carmona, Shanshan.Jiang, Bin.Dong}@cn.ricoh.com
takeshi.suzuki@jp.ricoh.com

Abstract

Domain adaptation of Large Language Models (LLMs) leads to models better suited for a particular domain by capturing patterns from domain text which leads to improvements in downstream tasks. To the naked eye, these improvements are visible; however, the patterns are not so. How can we know which patterns and how much they contribute to changes in downstream scores? Through a Multilevel Analysis we discover and quantify the effect of text patterns on downstream scores of domain-adapted Llama 2 for the task of sentence similarity (BIOSSES dataset). We show that text patterns from PubMed abstracts such as clear writing and simplicity, as well as the amount of biomedical information, are the key for improving downstream scores. Also, we show how another factor not usually quantified contributes equally to downstream scores: choice of hyperparameters for both domain adaptation and fine-tuning.

1 Introduction

Domain Adaptive Pretraining (DAPT) is an effective method to adapt a model to a particular domain via continual pretraining (Gururangan et al., 2020; Rietzler et al., 2020), with BioBERT (Lee et al., 2019) being a successful case in the biomedical domain. From our side, we have widely used DAPT not only to adapt LLMs to biomedicine, but as a part of a bigger pipeline to create customer-oriented, intelligent agents which can faithfully recover domain knowledge from both their parameters and external databases. However, domain adapting Llama 2 (Touvron et al., 2023) brought us a puzzle: huge variability on downstream scores.

Given both the size of Llama 2 and GPU memory restrictions, the domain adaptation of Llama 2 was restricted to a sample (subset) of PubMed abstracts. After domain-adapting and fine-tuning Llama 2 on PubMed abstracts and on the BIOSSES dataset (Soğancıoğlu et al., 2017), respectively, we obtained highly different downstream scores depending on the choice of the sample used for DAPT, as displayed in Fig. 1. This figure shows a huge variability in Pearson (downstream) scores: from 67 points up to an almost perfect score of 98 points. Surely, hyperparameter choice for both DAPT and fine-tuning has an impact on the scores, but, by comparing the patterns of score variation across samples used for DAPT we observe that this variation does not seem to be explained only by the choice of hyperparameter values. Clearly, data used for DAPT has also an impact on the scores.

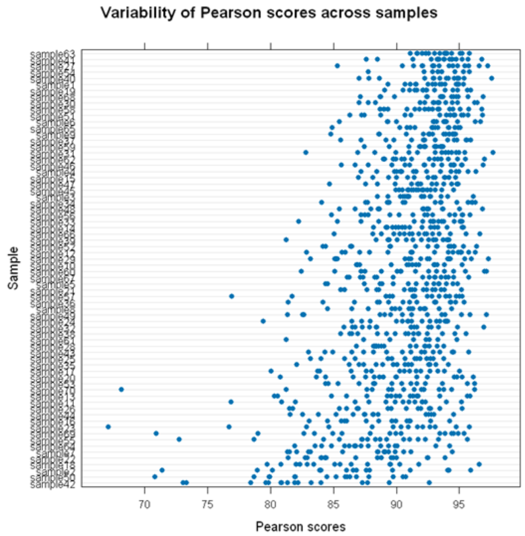


Figure 1: Variations in downstream score depending on the choice of both sample used for DAPT and hyperparameters. Each row represents a sample; each dot a Pearson score from a fine-tuned model (BIOSSES downstream dataset).

Thus, we asked: What features from the samples used for DAPT impact on the downstream scores and to what extent? And to what extent is the impact of the choice of hyperparameter values? We hypothesized that text patterns, such as sentence length, syntactic dependencies, or text complexity, among others, impact on the downstream scores. In order to analyze the importance of each text feature, and the effect of hyperparameters, we used Multilevel Modeling, a regression model widely used in the Social Sciences to explain social phenomena such as the effect of both school and student features on the students' performance scores.

Our results are simple: clarity and simplicity of writing, and amount of biomedical information, are key text features from PubMed abstracts for improving downstream scores. Moreover, variation in scores is also largely due to the choice of hyperparameters, contributing approximately as equal as the text features. These results not only explain important features from domain adaptation but can also serve for designing better document sampling strategies and hyperparameter search methods. Moreover, the use of Multilevel Models (MLMs) is key for a deeper understanding of NLP models which we hope the NLP community will adopt.

2 Related Work

2.1 Analyses of Large Language Models

Different works have analyzed different aspects of LLMs. For example, some works studied the interplay between data and model abilities during the SFT (Supervised Fine-Tuning) phase showing the key impact of data on these abilities (Dong et al., 2024). Other works analyzed which training instances contributed to specific model predictions, or abilities learned, using methods such as gradient-based tracing-back (Koh and Liang, 2017; Garima et al., 2020; Akyurek et al., 2022) and machine unlearning (Jang et al., 2023; Eldan and Russinovich, 2023; Zhao et al., 2024). Furthermore, strategies have been proposed to improve both the quality and selection of pretraining data to optimize LLMs' training time, perplexity, or capabilities on downstream tasks (Lee et al., 2022; Rae et al., 2022; Tirumala et al., 2023; Nguyen et al., 2023). However, to our knowledge, ours is the first work analyzing the effect of text features from samples used for domain adaptation on downstream scores via Multilevel Analysis.

2.2 Multilevel Modeling

Multilevel Models (MLMs) are extensively used across fields in the Social Sciences to measure the effect of multi-level variables on an outcome. For example, in Education, MLMs can predict student performance while finding out the most important features from students (level-1) and schools (level-2) (Rasbash et al., 2010; Goldstein et al., 2007); also, MLMs are used to compare school effectiveness (Yang et al., 2002; Goldstein et al., 1993). In Epidemiology, MLMs have been used to 1) model the impact of personal-level risk factors on disease across populations (Weinmayr et al., 2016); 2) measure the effect of air pollution on cardiovascular disease (Forbes et al., 2009); and 3) estimate the risks of food constituents from different items for breast cancer (Witte et al., 1994).

3 Data and Multilevel Model

3.1 Multilevel Regression Analysis

In MLMs, the dependent variable (downstream scores) depends on a set of independent variables which can be at different levels in a hierarchy. We model our problem as a 2-level hierarchy where features from DAPT and fine-tuning, such as choice of hyperparameters, correspond to level-1 variables; and text features from the samples used for domain adaptation of Llama 2 correspond to level-2 variables. This choice of level-1 and level-2 variables is due to our design of the domain adaptation and fine-tuning of Llama 2: from each sample, by varying DAPT hyperparameter values, we obtain 2 domain-adapted models, and from each of these two models, by varying fine-tuning hyperparameters, we obtain 8 fine-tuned models; i.e., from each sample we obtain 16 fine-tuned models; from the perspective of Multilevel Analysis, we say that each sample is a group and the 16 fine-tuned models are grouped under this sample.[1]

Thus, variables at level 1 are indicator variables signaling the use of a specific combination of hyperparameters for DAPT and fine-tuning where values of random seed, batch size, among other hyperparameters, vary; and level-2 variables correspond to numeric and indicator variables capturing text features (Section 3.2). Then, a 2-level MLM (Hox et al., 2017) can be expressed as:

$$y = \beta_0 + u_{0j} + \beta_1 x_1 + ... + \beta_n x_n \\ + \gamma_{1j} x_1 + ... + \gamma_{kj} x_k + e \quad (1)$$

where β_0 is the intercept and u_{0j} is a term called *random intercepts* which can be interpreted as a

[1] We chose MLM over simple linear regression since it is designed to deal with grouped (non-independent) instances while allowing for multi-level variables.

deviation in downstream score from the intercept according to each sample j; level-1 and level-2 variables are denoted by x_i and the terms β_i are called *fixed-effects* coefficients which represent the average effect of each variable (across all samples) on downstream scores (y); terms γ_{ij}, called *random coefficients or slopes*, are key terms in Multilevel Analysis and can be interpreted as an *adjusted* effect on the level-1 fixed-effects coefficients (β_i) according to each sample j;[2] e is the residual.

The advantage of modeling random coefficients lies in capturing differential contributions of each sample on downstream scores; that is, we expect different combinations of hyperparameters to have different effects, due to chance, on the scores depending on the choice of sample; thus, for each sample we can estimate the number of points (γ_{ij}) that a combination of hyperparameters deviates from the mean effect across all samples (β_i).

3.2 Data for Multilevel Regression

To fit an MLM that predicts downstream scores based on both text features from samples used for domain adaptation and choice of hyperparameters for DAPT and fine-tuning, we extract text features from 70 samples[3] used for domain-adapting Llama 2 and use indicator variables to signal the use of a specific hyperparameter combination.

We obtained 1120 fine-tuned models but 16 were discarded since the outputs generated were outside the set of permissible outputs (a ranking score between 0 and 4 showing the degree of similarity between two input sentences), so we used 1104 models. Each fine-tuned model corresponds to an instance in our dataset for fitting MLMs. The dependent variable corresponds to the Pearson correlation score (scaled to [0-100] points) of a model's outputs with gold outputs from the BIOSSES validation set (as measured in the BLURB benchmark (Gu et al., 2021)). Independent variables correspond to features at levels 1 and 2 which we group in 8 groups according to their type. These text features are motivated by research on Education for predicting student writing performance since they have been shown to be strong predictors. We describe these features:

Hyperparameter choice: Level-1 indicator variables signaling the choice of hyperparameter combination used for both domain adaptation and fine-tuning. In Section 4.2 we call these variables DAPT_1 , DAPT_2 (2 combinations for DAPT) and HCF_1, ..., HCF_8 (8 combinations for fine-tuning).

Syntactic dependencies: We extracted 39 syntactic relations from the sentences in each sample via the Stanford dependency parser (Chen and Manning, 2014) and we used the frequency of each relation across the whole sample as a level-2 feature.

Terms overlaps: We hypothesized that overlap of information contained in the sample used for DAPT with information in the BIOSSES dataset may help to improve downstream scores; thus, we computed the frequency of overlapping terms. To do so, we computed frequencies of biomedical and non-biomedical terms at the unigram and bigram levels separately for the train and validation sets of the BIOSSES dataset (leading to eight level-2 numeric features) using the frequency metric of Kerz et al. (2021).

Biomedical information: We computed the ratio of biomedical unigrams to the total number of terms occurring in each sample used for DAPT as we hypothesized that the more the amount of biomedical terms in a sample the better the downstream scores; this led to a numeric level-2 feature.

Text complexity: We measured the linguistic complexity of the samples at 3 different levels: morphological, syntactical, and global, according to Kolmogorov metrics of complexity used in linguistics (Ehret and Szmrecsanyi, 2019), leading to three level-2 features. We hypothesized that complex texts may provide more information and thus better scores.

Average lengths: From each sample, we computed average lengths of both PubMed abstracts (in terms of words) and words (in terms of characters) resulting in two level-2 features.

Sample size: We hypothesized that the number of PubMed abstracts matter, so we tried two different sample sizes for DAPT: 25K and 50K, operationalized as a level-2 indicator variable.

Sampling method: We hypothesized that sampling contiguous abstracts, in terms of publication time, could improve scores since biomedical information tend to be more uniform; thus, we tried

[2] This only applies to level-1 variables, thus $k < n$ (Eq. 1).

[3] The suggested minimum number of groups is 50 (Maas and Hox, 2005).

two sampling methods: randomly and contiguously, which we operationalized as a level-2 variable.

4 Multilevel Analysis and Results

4.1 Fitting Multilevel Models

Goals: We have 2 goals (Harrell, 2015). First, finding which variables have a statistically-significant effect on downstream scores. And second, evaluating the predictive behavior of our Multilevel Model to unseen data via cross-validation.

Modeling Strategies: Our strategy is three-fold. First, we deal with the issue of *multicollinearity*, where it is difficult to assess the effect of variables when they are correlated, via a variant of the variable selection strategy from Yu et al. (2015). Second, we aim for a *parsimonious* model (Robson and Pevalin, 2016) that is simple enough, in the number of parameters, to be understood, yet complex enough to have low prediction error. Third, we perform suggested evaluations in the literature such as likelihood ratio tests (Brown, 2021), R-squared effects (Rights and Sterba, 2019), and cross-validation (Lindner et al., 2022).[4] Lastly, we note that we standardize (mean=0, std dev=1) all independent variables to allow for a head-to-head comparison of their impact on downstream scores.

The curse of multicollinearity: We found extreme cases of multicollinearity across most variables (Fig. 2). To alleviate this problem, we adjust the strategy of Yu et al. (2015): we first use lasso to eliminate non-essential variables; then, we discard redundant variables via variance decomposition proportions; and finally, we apply a backwards search to remove non-significant variables. To avoid introducing bias, we confirm our choice of deletion by measuring cross-validation error.

4.2 Regression Results

We show the results of our best MLM: we show the features that have a significant impact on scores. For Tables 1 and 2, the statistical significance code is: p=0 '***', p<0.001 '**', p<0.01 '*'.

Biomedical information matters: In Table 1 we observe the standardized coefficient of Biomedical_info having a positive and statistically significant effect of 1.78 meaning that for every standard deviation increase in the frequency of biomedical terms in a sample used for DAPT, the downstream scores increase, in average, by 1.78 points.

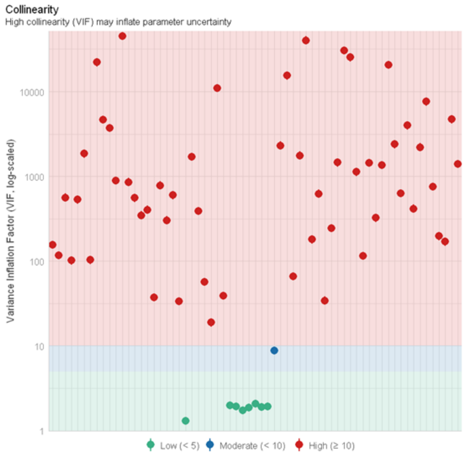


Figure 2: Multicollinearity of all independent variables according to Variance Inflation Factor scores. Scores bigger than 10 show severe cases of multicollinearity.

Text structure and clarity of writing matters: As we observe in Table 1, the syntactic dependency of parataxis has a negative effect on downstream scores: for every standard deviation increase in frequency, scores reduce by 1.92 points. Parataxis occurs when complex sentences are split into clauses separated by commas or semicolons without using any subordinating or coordinating conjunction to make their relationship clear (de Marneffe et al., 2021). Academic writing, as that in PubMed abstracts, often uses parataxis, for example, for reporting previous findings. If overused, parataxis can convey a sense of text unclarity. In contrast, adnominal clauses (acl) occur when the main nominal in a sentence is modified by a subordinate clause usually via clear connectors and in a specific order which conveys text clarity. As we observe in Table 1, acl is the only syntactic relation having a positive effect on downstream scores.

Simplicity matters: As shown in Table 1, two other dependencies have a negative effect on scores: mwe (multiword expressions) and cc.preconj. Simply stated, complex terms such as compounds (e.g. *USB cellphone charger*), proper names, fixed expressions (e.g. *as well as*), or preconjuncts (e.g. *both DNA and RNA*) (de Marneffe et al., 2021), which are common in academic writing, decrease scores. Moreover, longer words tend to substantially decrease scores, as captured by the feature Avg_word being, surprisingly, the feature with the biggest negative impact. Thus, a concise writing

[4] We compute 5-fold cross-validated RMSE (Root Mean Squared Error), averaged over 5 different random seeds.

Variable	Coeff. (β)	SE	t
Intercept	92.66***	0.28	325.58
DAPT_1	-1.00*	0.43	-2.30
HCF_1	-2.53***	0.38	-6.63
HCF_2	-2.49***	0.34	-7.20
HCF_3	-1.12**	0.35	-3.20
HCF_4	-1.12***	0.29	-3.75
HCF_5	-2.76***	0.37	-7.42
HCF_6	-2.68***	0.37	-7.25
acl	1.86**	0.59	3.15
mwe	-0.82**	0.28	-2.87
parataxis	-1.92**	0.56	-3.39
cc.preconj	-3.04***	0.72	-4.20
Biomedical_info	1.78*	0.71	2.49
Avg_word	-3.91***	0.80	-4.84

Table 1: Results of MLM: fixed-effects of variables at levels 1 and 2. Coeff: coefficient. SE: Standard Error. t: t-value (values truncated at the hundredths). DAPT_1 and HCF_i are indicator variables signaling the use of a specific combination of hyperparameters for DAPT and fine-tuning, respectively. We use DAPT_2 and HCF_8 as references to avoid perfect collinearity.

Variable	Variance	Std. Dev.
Intercepts	3.06***	1.75
HCF_1	3.90***	1.97
HCF_2	2.09*	1.44
HCF_3	2.34**	1.53
HCF_5	3.38**	1.84
HCF_6	3.28***	1.81
DAPT_1	11.03***	3.32

Table 2: Results of MLM: random-effects (random intercepts and random coefficients). We use DAPT_2 and HCF_8 as references to avoid perfect collinearity.

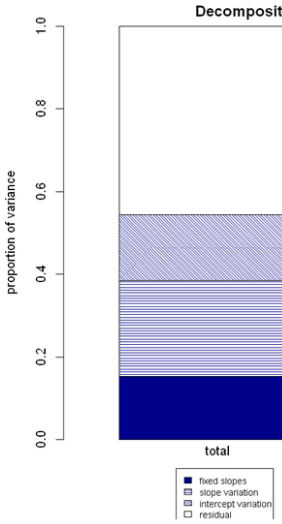


Figure 3: R-squared: Decomposition of variance across fixed and random effects.

with less *idiomatic* and complex expressions is key for a better domain adaptation. However, it is unclear whether biomedical terms, which are often complex, may jeopardize the domain adaptation; thus, this phenomenon deserves a deeper analysis for future work.

How much hyperparameters impact on scores? As we see in Table 1, different hyperparameter combinations lead to different results being combination HCF_5 the one with the biggest impact: whenever used, it leads to an average decrease of 2.76 points in scores across all samples. Moreover, in Table 2 we observe that the choice of sample adds a random effect to the fixed effect of most of the hyperparameter combinations; i.e., we can expect average variations of (±)[1.44-1.97] and (±) 3.32 points in the effects of fine-tuning and DAPT hyperparameters observed in Table 1, respectively.

Cross-validation error: Our MLM obtains an RMSE of only 4.02 points, which means that our model will deviate, in average, only by 4 points from expected Pearson scores on unseen data.

R-squared effects: Figure 3 shows that around 55% of the variation in downstream scores is accounted by fixed- and random-effects, where 15% is due to fixed-effects, 23% due to random-effects (slopes), and the rest (intercept variation) is due to other features from the samples that we were not able to identify. From Fig. 3 and Table 1, we estimate that the overall effect of hyperparameters on downstream scores is approx. equal to that of text features for domain adaptation described above.

5 Conclusions

How important is to analyze the data used for DAPT? Working *in the trenches* has allowed us to see the paramount importance that data plays on the right adjustment of LLMs to a target domain. From a customer-oriented perspective, DAPT plays a vital role for the correct adjustment of LLMs not only to parametric knowledge but also to human alignment via SFT and to databases via RAG (Retrieval Augmented Generation). Thus, as in a *snowball effect*, studying the factors that matter for biomedical DAPT –clarity and simplicity of writing as well as biomedical information– assures us to provide better adapted LLMs for customer applications.

Limitations

We note that our work has some limitations. For example, despite the low RMSE from our MLM (4% error) and even though we tried our best to propose a comprehensive set of variables that could fully explain the variation in downstream scores, we acknowledge (as observed in Fig. 3) that 45% of the variation in scores remains unexplained; that is, there are more variables at level 1 and level 2 that may have an impact on downstream scores. Furthermore, while we chose the most widely used type of MLM in the literature (2-level MLM), it is possible that other choice of MLM may be a better fit to our problem such as a 3-level model where at level-1 we define only hyperparameter combinations of DAPT, at level-2 we define hyperparameter combinations of fine-tuning, and at level-3 we define features from the samples; however, a model of this type requires a substantial increase in the number of both domain-adapted and fine-tuned models and thus of computing time.

Acknowledgments

We thank the anonymous reviewers for their thorough comments and to Zhang Yuming for sharing Llama 2 with us.

References

Ekin Akyurek, Tolga Bolukbasi, Frederick Liu, Binbin Xiong, Ian Tenney, Jacob Andreas, and Kelvin Guu. 2022. Towards tracing knowledge in language models back to the training data. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2429–2446, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Violet A. Brown. 2021. An introduction to linear mixed-effects modeling in R. *Advances in Methods and Practices in Psychological Science*, 4(1):1–19.

Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 740–750, Doha, Qatar. Association for Computational Linguistics.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2024. Qlora: efficient finetuning of quantized llms. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. 2024. How abilities in large language models are affected by supervised fine-tuning data composition. *ArXiv*.

Katharina Ehret and Benedikt Szmrecsanyi. 2019. Compressing learner language: An information-theoretic measure of complexity in sla production data. *Second Language Research*, 35(1):23–45.

Ronen Eldan and Mark Russinovich. 2023. Who's harry potter? approximate unlearning in llms. *ArXiv*.

Lindsay J. L. Forbes, Minal D. Patel, Alicja R. Rudnicka, Derek G. Cook, Tony Bush, John R. Stedman, Peter H. Whincup, David P. Strachan, and Ross H. Anderson. 2009. Chronic exposure to outdoor air pollution and markers of systemic inflammation. *Epidemiology*, 20(2):245–253.

Garima, Frederick Liu, Satyen Kale, and Mukund Sundararajan. 2020. Estimating training data influence by tracing gradient descent. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Harvey Goldstein, Simon Burgess, and Brendon McConnell. 2007. Modelling the effect of pupil mobility on school differences in educational achievement. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, 170(4):941–954.

Harvey Goldstein, Jon Rasbash, Min Yang, Geoffrey Woodhouse, Huiqi Pan, Desmond Nuttall, and Sally Thomas. 1993. A multilevel analysis of school examination results. *Oxford Review of Education*, 19(4):425–433.

Andreas Groll. 2023. *glmmLasso: Variable Selection for Generalized Linear Mixed Models by L1-Penalized Estimation*. R package version 1.6.3.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare*, 3(1).

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Jr. Frank E. Harrell. 2015. *Regression Modeling Strategies: With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis*, second edition. Springer Cham.

J. Hox, M. Moerbeek, and R. van de Schoot. 2017. *Multilevel Analysis: Techniques and Applications*, third edition. Routledge.

Muhammad Imdadullah, Muhammad Aslam, and Saima Altaf. 2016. mctest: An R Package for Detection of Collinearity among Regressors. *The R Journal*, 8(2):495–505.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2023. Knowledge unlearning for mitigating privacy risks in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14389–14408, Toronto, Canada. Association for Computational Linguistics.

Elma Kerz, Daniel Wiechmann, Yu Qiao, Emma Tseng, and Marcus Ströbel. 2021. Automated classification of written proficiency levels on the CEFR-scale through complexity contours and RNNs. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 199–209, Online. Association for Computational Linguistics.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1885–1894. JMLR.org.

Alexandra Kuznetsova, Per B. Brockhoff, and Rune H. B. Christensen. 2017. lmerTest package: Tests in linear mixed effects models. *Journal of Statistical Software*, 82(13):1–26.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022. Deduplicating training data makes language models better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8424–8445, Dublin, Ireland. Association for Computational Linguistics.

Thomas Lindner, Jonas Puck, and Alain Verbeke. 2022. Beyond addressing multicollinearity: Robust quantitative analysis and machine learning in international business research. *Journal of International Business Studies*, 53(7).

Daniel Lüdecke, Mattan S. Ben-Shachar, Indrajeet Patil, Philip Waggoner, and Dominique Makowski. 2021. performance: An R package for assessment, comparison and testing of statistical models. *Journal of Open Source Software*, 6(60):3139.

Cora J. M. Maas and Joop J. Hox. 2005. Sufficient sample sizes for multilevel modeling. *Methodology: European Journal of Research Methods for The Behavioral and Social Sciences*, 1(3):86–92.

Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2023. Culturax: A cleaned, enormous, and multilingual dataset for large language models in 167 languages. *ArXiv*.

Ludvig Renbo Olsen and Hugh Benjamin Zachariae. 2023. *cvms: Cross-Validation for Model Selection*. R package version 1.6.0.

Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu, Erich Elsen, Siddhant Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew Johnson, Blake Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Ed Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorrayne Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. 2022. Scaling language models: Methods, analysis & insights from training gopher. *ArXiv*.

Jon Rasbash, George Leckie, Rebecca Pillinger, and Jennifer Jenkins. 2010. Children's Educational Progress: Partitioning Family, School and Area Effects. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 173(3):657–682.

Alexander Rietzler, Sebastian Stabinger, Paul Opitz, and Stefan Engl. 2020. Adapt or get left behind: Domain adaptation through BERT language model finetuning for aspect-target sentiment classification. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4933–4941, Marseille, France. European Language Resources Association.

J. D. Rights and S. K. Sterba. 2019. Quantifying explained variance in multilevel models: An integrative framework for defining r-squared measures. *Psychological Methods*, 24(3):309–338.

Karen Robson and David Pevalin. 2016. *Multilevel Modeling in Plain Language*, first edition. SAGE Publications Ltd.

Deepayan Sarkar. 2008. *Lattice: Multivariate Data Visualization with R*. Springer, New York.

Mairead Shaw, Jason D. Rights, Sonya S. Sterba, and Jessica Kay Flake. 2022. r2mlm: An r package calculating r-squared measures for multilevel models. *Behavior Research Methods*, 55:1942–1964.

Gizem Soğancıoğlu, Hakime Öztürk, and Arzucan Özgür. 2017. BIOSSES: a semantic sentence similarity estimation system for the biomedical domain. *Bioinformatics*, 33(14):i49–i58.

Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari Morcos. 2023. D4: Improving llm pretraining via document de-duplication and diversification. In *Advances in Neural Information Processing Systems*, volume 36, pages 53983–53995. Curran Associates, Inc.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *ArXiv*.

Gudrun Weinmayr, Jens Dreyhaupt, Andrea Jaensch, Francesco Forastiere, and David P Strachan. 2016. Multilevel regression modelling to investigate variation in disease prevalence across locations. *International Journal of Epidemiology*, 46(1):336–347.

J. S. Witte, S. Greenland, R. W. Haile, and C. L. Bird. 1994. Hierarchical regression analysis applied to a study of multiple dietary exposures and breast cancer. *Epidemiology*, 5(6):612–621.

Min Yang, Harvey Goldstein, William Browne, and Geoffrey Woodhouse. 2002. Multivariate Multilevel Analyses of Examination Results. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 165(1):137–153.

Han Yu, Shanhe Jiang, and Kenneth C. Land. 2015. Multicollinearity in hierarchical linear models. *Social Science Research*, 53:118–136.

Yang Zhao, LI DU, Xiao Ding, Kai Xiong, Zhouhao Sun, Jun Shi, Ting Liu, and Bing Qin. 2024. Deciphering the impact of pretraining data on large language models through machine unlearning. *ArXiv*.

A Appendix

A.1 Statistical Software

To fit MLMs we use the R-package *lmerTest* (Kuznetsova et al., 2017). To compute cross-validated RMSE we use the *cvms* package (Olsen and Zachariae, 2023), where for folds used as test data, we leave out all fine-tuned models from the samples selected for testing to avoid training and testing on models derived from the same sample. Furthermore, to estimate the proportion of explained variability in downstream scores we compute R-squared effects using the framework of Rights and Sterba (2019) via the R-package *r2mlm* (Shaw et al., 2022). We plot Figures 1 and 2 via the *lattice* (Sarkar, 2008) and *performance* (Lüdecke et al., 2021) packages in R, respectively. To fit lasso regression we use the *glmmLasso* (Groll, 2023) package in R; to compute variance decomposition proportions we use the *mctest* (Imdadullah et al., 2016) package in R. Likelihood ratio tests and backward search are implemented via the *lmerTest* package.

A.2 Training Features

We used a Titan RTX GPU (24GB of memory) for both domain adaptive pretraining and fine-tuning. We domain-adapted Llama 2 for 1 epoch with each sample. We fine-tuned each domain-adapted model with each hyperparameter combination for 30 epochs and kept models with the highest validation score. We used QLoRA (Dettmers et al., 2024) to be able to fit Llama 2 in GPU memory.

Mention-Agnostic Information Extraction for Ontological Annotation of Biomedical Articles

Oumaima El Khettari[1,2*] **Noriki Nishida**[2*] **Shanshan Liu**[2] **Rumana Ferdous Munne**[2]
Yuki Yamagata[3,4] **Solen Quiniou**[1] **Samuel Chaffron**[1] **Yuji Matsumoto**[2]
[1]Nantes Université - LS2N [2]RIKEN AIP [3]RIKEN R-IH [4]RIKEN BRC
{oumaima.el-khettari, solen.quiniou, samuel.chaffron}@univ-nantes.fr
{noriki.nishida, shanshan.liu, rumanaferdous.munne,
yuki.yamagata, yuji.matsumoto}@riken.jp

Abstract

Biomedical information extraction is crucial for advancing research, enhancing healthcare, and discovering treatments by efficiently analyzing extensive data. Given the extensive amount of biomedical data available, automated information extraction methods are necessary due to manual extraction's labor-intensive, expertise-dependent, and costly nature. In this paper, we propose a novel two-stage system for information extraction where we annotate biomedical articles based on a specific ontology (HOIP). The major challenge is annotating relation between biomedical processes often not explicitly mentioned in text articles. Here, we first predict the candidate processes and then determine the relationships between these processes without relying on mentions. The experimental results show promising outcomes in mention-agnostic process identification using Large Language Models (LLMs). In relation classification, our proposed BERT-based models outperform LLMs significantly. The end-to-end evaluation results suggest the difficulty of this task and room for improvement in both process identification and relation classification.

1 Introduction

In the biomedical domain, unraveling the mechanisms underlying various diseases contributes significantly to their treatment and prevention. However, information about these mechanisms is often scattered across articles, presenting challenges. The challenges include the lack of clarity, the implicit nature of background knowledge, and the ad hoc use of vocabularies with variations in notation. Moreover, inherent biological complexity spans molecules, cells, and organs, with external factors such as viruses influencing infection mechanisms.

To address these challenges, organizing knowledge through ontologies is crucial as they provide a clear framework for consistently structuring entities and their relationships. In the Homeostasis Imbalance Process Ontology (HOIP), manual annotation has been employed to extract and structure knowledge about processes such as cellular senescence and COVID-19 infection mechanisms (Yamagata et al., 2021, 2024). Despite these systematic approaches, manual annotation faces significant challenges due to its high cost and time-consuming nature. These challenges highlight the need for more efficient and consistent (semi-)automated annotation approaches to improve the overall quality and usefulness of ontologies.

In this paper, we propose an application of Natural Language Processing (NLP) as a promising solution. Specifically, assuming automatic annotation of the HOIP ontology as our ultimate goal, we propose a two-stage information extraction (IE) system. Figure 1 shows an overview of our two-stage system. Given an input passage[1], the first stage, *Process Identification*, identifies process entities that are described in the passage or can be inferred using the domain knowledge.[2] The entities are represented as unique IDs in the ontology. The entities are then passed to the second stage, *Document-level Relation Extraction (DocRE)* (Christopoulou et al., 2019; Zhou et al., 2021; Xiao et al., 2022; Zhang et al., 2021; Li et al., 2023), to classify entity pairs into a pre-defined set of interrelations. The system output is represented as a set of triples: {(head entity ID, relation, tail entity ID)}. We develop and evaluate different approaches including supervised models based on BERT (Devlin et al., 2019) and

[*] Equal contribution.

[1] In this paper, we use the word "passage" instead of "document" or "paragraph" to describe the input in our task, because the text describing biomedical processes is not necessarily a complete text like an entire paragraph or document.

[2] Process Identification is similar to Entity Disambiguation (ED), but differs as discussed in the following paragraph. Given an input passage, ED aims to identify entities (IDs) for each given mention, whereas Process Identification aims to identify entities (IDs) without the availability of mentions.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 457–473
August 16, 2024. ©2024 Association for Computational Linguistics

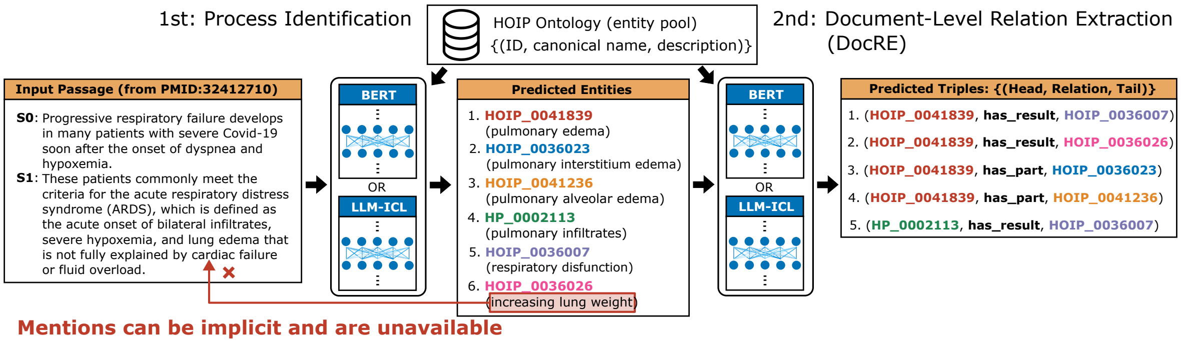

1st: Process Identification
HOIP Ontology (entity pool)
{(ID, canonical name, description)}
2nd: Document-Level Relation Extraction (DocRE)
Input Passage (from PMID:32412710)
S0: Progressive respiratory failure develops in many patients with severe Covid-19 soon after the onset of dyspnea and hypoxemia.
S1: These patients commonly meet the criteria for the acute respiratory distress syndrome (ARDS), which is defined as the acute onset of bilateral infiltrates, severe hypoxemia, and lung edema that is not fully explained by cardiac failure or fluid overload.
BERT
OR
LLM-ICL
Predicted Entities
1. HOIP_0041839 (pulmonary edema)
2. HOIP_0036023 (pulmonary interstitium edema)
3. HOIP_0041236 (pulmonary alveolar edema)
4. HP_0002113 (pulmonary infiltrates)
5. HOIP_0036007 (respiratory disfunction)
6. HOIP_0036026 (increasing lung weight)
BERT
OR
LLM-ICL
Predicted Triples: {(Head, Relation, Tail)}
1. (HOIP_0041839, has_result, HOIP_0036007)
2. (HOIP_0041839, has_result, HOIP_0036026)
3. (HOIP_0041839, has_part, HOIP_0036023)
4. (HOIP_0041839, has_part, HOIP_0041236)
5. (HP_0002113, has_result, HOIP_0036007)
Mentions can be implicit and are unavailable

Figure 1: An overview of our mention-agnostic two-stage information extraction system with a real example in our HOIP dataset. Given an input passage, the first stage identifies process entities described in the passage or inferable based on the domain knowledge. The predicted entities are then passed to the second stage to identify relations between them. Please note that our system does not rely on mentions, enabling extraction of structured knowledge about entities and relations described implicitly in the passage.

generative methods based on Large Language Models (LLMs) and In-Context Learning (ICL) (Brown et al., 2020; Chowdhery et al., 2022; Wadhwa et al., 2023; Ozyurt et al., 2024) for both process identification and DocRE. The HOIP dataset, a novel manually annotated dataset built based on the HOIP ontology for biomedical IE system development, will be available to the public.

Traditional IE studies (Yu et al., 2020; Wu et al., 2020; Zhou et al., 2021) assume that an entity can appear multiple times in a passage **explicitly** (such textual instances are called *mentions*), and derive entity features from these mentions. Mentions are strong indicators in IE, since they directly indicate how entities are described in a text. However, in real-world scenarios including our HOIP dataset, an entity sometimes appears only **implicitly**. With no availability of mentions, it is not obvious how to induce useful entity features from a passage. This paper proposes multiple approaches that do not require explicit mentions.

Our contributions and findings are summarized as follows:

- We release the HOIP dataset, to facilitate the development and bench-marking of IE models for the real-world ontology.
- We develop a mention-agnostic two-stage IE system, which enables to extract structured knowledge described implicitly in text. BERT-based supervised models and LLM-based models are presented for both process identification and DocRE.
- Experimental results in process identification suggest that generative models are valuable for low-resource in-domain corpora like the HOIP dataset.
- DocRE results suggest that, although mentions are strong indicators, the proposed BERT-based models outperform LLMs and achieve F1 scores of around 56-59 points even without mention hints.
- Evaluation results on the end-to-end system reveal that improvements in both process identification and DocRE are crucial in the current stage.
- The HOIP dataset and the source codes are available: https://github.com/norikinishida/hoip-dataset (dataset), https://github.com/sl-633/bio-process-identifier (process identification), https://github.com/norikinishida/kapipe (DocRE).

2 HOIP Dataset

Our ultimate goal is to update and improve ontologies by (semi-)automatically extracting entities and their interrelations from articles. As a testbed ontology, we choose the Homeostasis Imbalance Process Ontology (HOIP) (Yamagata et al., 2021, 2024), which focuses on understanding the COVID-19 infectious mechanism (courses).[3] To facilitate the development of NLP systems and benchmark the task, we construct and release a new dataset named the HOIP dataset based on the HOIP ontology. The dataset includes passages extracted from PubMed articles describing biomedical processes in the context of COVID-19 infectious courses. Each passage is a brief portion of a PubMed article that describes at least two specific processes. The processes are manually annotated as a set of triples, i.e., {(head entity, relation, tail entity)}. Figure 1 shows a real example in the dataset.

[3] For the details of the ontology see Appendix A.

	Train	Dev	Test
# passages	255	35	37
# entities	1988	143	211
# triples	1848	137	177
Avg. words per passage	75.5	70.4	61.8
Avg. entities per passage	7.8	4.1	5.7
Avg. triples per passage	7.2	3.9	4.8

Table 1: Dataset statistics for the HOIP dataset.

2.1 Data Collection and Enhancement

We first stored the HOIP ontology files in an RDF store using Apache Jena Fuseki[4], and constructed a SPARQL endpoint. We used SPARQL queries to retrieve the information required for the dataset. The results were then converted to CSV. To optimize the dataset for machine annotation and enhance its clarity, we made several adjustments based on the hierarchical structure of the HOIP ontology. Originally, some process entities included *course* information, such as "blood vessel damage *in severe COVID-19*" (the course is in italics), indicating a specific context. We removed these course information from the entities to optimize for machine annotation. Additionally, processes with too fine granularity were deemed unsuitable for machine annotation predictions. Therefore, we prioritized processes that are generalized using superclasses of each process, assigning Gene Ontology (GO) terms (Ashburner et al., 2000). This approach ensures that the annotations are practical for applications for reusability.

2.2 Dataset Organization

In the CSV file generated by the above procedure, each record corresponds to one triple. We combined the triples associated with the same passage (string) and PubMed ID into the same group, and this group was considered to be a single example in the final dataset. We found that there were textual overlaps across the passages. Thus, if a passage d_{src} was textually contained in another passage d_{dst} and both passages are associated with the same PubMed ID, the triples T_{src} for d_{src} were merged into the triples T_{dst} for d_{dst}. Finally, we split the entire dataset into training, development, and test sets, ensuring that passages extracted from the same article were not scattered across different splits. The dataset statistics are shown in Table 1.

3 Methods for Process Identification

In the HOIP dataset, a biological process entity is annotated depending on whether it is mentioned (explicitly or implicitly) in the passage, without specifying the corresponding phrase of the entity in the passage. This makes the dataset more closely match the real-world scenario, but also brings challenges to the automatic process identification – directly employing Named Entity Recognition methods that require the correspondence between a explicit mention (entity text and offsets) and an input text for model training is no longer an option. To address this task, we propose approaches to identify biological processes without prior recognition of mentions that can be matched to terminological expressions of entities in the HOIP ontology. Two distinct approaches are developed: BERT-based supervised methods and LLM-based In-Context Learning (ICL) methods.

3.1 BERT-based Supervised Approach

Considering that 360 unique process names are encompassed in the HOIP dataset, the task of process identification can be approached as a multi-class and multi-label classification problem. For simplification purposes, we convert the task into a binary format, framing it in the following manner: Let D be the set of passages and A be the set of annotated process names. The input sequence is constructed for each passage from D as follows:

$$[\texttt{CLS}]\ \text{passage}\ [\texttt{SEP}]\ \text{name}\ [\texttt{SEP}]$$

where name denotes a process name $a_i \in A$. Then, the task is a binary classification task whether the passage involves the process or not.

3.2 LLM-based ICL Approach

Taking into account the unique characteristics of the dataset and the rapid advancements in the capabilities of LLMs to produce coherent texts in low resource settings (Wang et al., 2023), LLMs are utilized in this study to generate HOIP processes for each passage. We aim to evaluate the model's performance in low-resource settings characterized by imbalanced data in a specialized domain, and assess the model's generative capability in producing HOIP ontology terms.

- Zero-shot setting: The model is prompted to list the biological processes present in the text. Following a prompt format being demonstrated effective in many studies (Mishra et al., 2022;

[4] https://jena.apache.org/documentation/fuseki2/

Sclar et al., 2023), the prompt includes task instruction, constraints on the output, the input text. An example of the prompt is shown in Table 8.

- Few-shot setting: Following the previously mentioned prompt format, two few-shot strategies are employed through adding demonstrations: the first involves selecting randomly three examples from the development set, while the second is selecting examples based on semantic closeness of process names.

4 Methods for Document-level RE

Given an input passage d, a set of entities for the passage $\{e_1, \cdots, e_K\}$, and a pre-defined set of relations $\mathcal{R}$, document-level relation extraction (DocRE) (Christopoulou et al., 2019; Zhou et al., 2021; Xiao et al., 2022; Zhang et al., 2021) aims to predict relations from $\mathcal{R} \cup \{\mathrm{NA}\}$ for entity pairs (e_i, e_j) $(i, j \in [1, K]; i \neq j)$, where e_i and e_j denote head and tail entities respectively, and the NA class indicates that the entity pair has no relation.

4.1 QA-Style DocRE Model

Our first approach is to perform DocRE as a Question Answering (QA) task. We first generate questions for each possible triple. The question and the input passage are concatenated and passed to a pre-trained language model for answering.

Question Generation. We first enumerate all possible entity pairs $\{(e_i, e_j)\}_{i,j\in[1,K];i\neq j}$, and then apply pre-defined template functions $\{\mathcal{T}_r\}_{r\in\mathcal{R}}$ to the entity pairs to obtain questions for *each* possible triple (e_i, r, e_j): $q_{(e_i,r,e_j)} = \mathcal{T}_r(e_i, e_j)$. Table 7 shows examples for the pre-defined templates. The input x to our QA model is as follows:

$$[\texttt{CLS}] \text{ question } [\texttt{SEP}] \text{ passage } [\texttt{SEP}]$$

where question and passage are the word-pieces tokens of $q_{(e_i,r,e_j)}$ and d, respectively. An example for the input is "[CLS] does immunoglobulin production result in immunoglobulin mediated immune response ? [SEP] within 19 days after sympton onset , 100 % ... [SEP]".

Answer Classification. Then, we feed the input sequence x into a BERT-based encoder (Devlin et al., 2019; Beltagy et al., 2019) to obtain the contextual embeddings: $\{\boldsymbol{h}_w\}_{w=1}^{N^x_{\mathrm{tok}}} = \mathrm{Encoder}(x)$, where N^x_{tok} is the number of tokens in x. We concatenate the output of the last layer for the [CLS] token and the average-pooling embedding to obtain the passage embedding: $\tilde{\boldsymbol{h}} = \boldsymbol{h}_1 \oplus \frac{1}{N^x_{\mathrm{tok}}} \sum_{w=1}^{N^x_{\mathrm{tok}}} \boldsymbol{h}_w$, where $\oplus$ represents the concatenation of vectors. Then, we apply a two-layer feed-forward network and sigmoid activation to the passage embedding to calculate the probability of answer "Yes".

Loss Function. The network is trained using a binary cross entropy loss to maximize the probability for the correct triples.

4.2 Mention-Agnostic ATLOP (MA-ATLOP)

Our first approach requires solving QAs for all the possible triples. Since the number of possible triples is increased by $O(K^2)$ for the number of entities K, this is not efficient. Our second approach is to make predictions over all possible triples in a single forward pass. We extend a traditional and popular DocRE method, ATLOP (Zhou et al., 2021), so as not to rely on explicit mentions. We call this method *Mention-Agnostic ATLOP*, or MA-ATLOP shortly.

Entity Encoding. We use a BERT-based encoder to encode each entity e_i and passage d jointly into a dense vector that takes into account how the entity e_i is described in the passage d. Specifically, we first retrieve the canonical names $\{n_i\}_{i=1}^K$ and the descriptions $\{s_i\}_{i=1}^K$ for the given entities from the ontology using the entity IDs as query. Then, for each entity e_i, we construct input x_i as follows:

$$[\texttt{CLS}] \text{ name : description } [\texttt{SEP}] \text{ passage } [\texttt{SEP}]$$

where name, description, and passage are the word-pieces tokens of n_i, s_i, and d, respectively. We apply the encoder to each input x_i independently to obtain contextual embeddings: $\{\boldsymbol{h}_{i,w}\}_{w=1}^{N^{x_i}_{\mathrm{tok}}} = \mathrm{Encoder}(x_i)$. We take the embedding of the [CLS] token as the entity embedding, i.e., $\boldsymbol{e}_i = \boldsymbol{h}_{i,1}$.

Relation Classification. After obtaining the entity embeddings $\{\boldsymbol{e}_i\}_{i=1}^K$, we apply two separate FFNNs and tanh activation to map them to different representations for the head/tail entities of triples. Then, we apply a group bilinear classifier (Zheng et al., 2019; Tang et al., 2020) to the head/tail representations of an entity pair (e_i, e_j). Specifically, we divide both head/tail representations into G contiguous groups and then apply bilinear to each group. They are then summed up to calculate the score for relation $r \in \mathcal{R} \cup \{\mathrm{TH}\}$. Refer to Zhou et al. (2021) for the detail of group bilinear. We follow ATLOP and employ the adaptive-thresholding class TH. The relations scored higher than the TH class are regarded as positive. If no such relation exists, the NA class is assigned to the entity pair.

Loss Function. We use the adaptive-thresholding loss proposed in ATLOP to push the scores of correct/incorrect relations to be higher/lower than the TH class.

Negative Entity Sampling (NES). In an experimental setting, we assume experts correctly annotate entities. However, in real-world situations, entities are automatically annotated by the systems. Thus, it often happens that entities not described in the passage are included in the given entity list $\{e_i\}_{i=1}^{K}$. Our DocRE system must be robust to such noisy (false-positive) entities. Therefore, we propose Negative Entity Sampling (NES), where we sample additional negative entities randomly and add them to the given entities $\{e_i\}_{i=1}^{K_{\text{pos}}}$ during training. We sample negative entities from all entities in the ontology. Given the number of positive entities K_{pos} and a hyperparameter $\rho > 0$, we define the number of sampled negative entities as $K_{\text{neg}} = \text{round}(\rho \times K_{\text{pos}})$, where round is the rounding function. For instance, for $K_{\text{pos}} = 10$ and $\rho = 0.5$, K_{neg} is 5. We add a linear layer to the network to classify whether the entity is described in the passage: $y_i^{\text{ent}} = \sigma(\text{FFNN}_{\text{ent}}(\boldsymbol{e}_i))$. We use a binary cross entropy as an auxiliary loss to maximize y_i^{ent} for positive entities.

4.3 LLM and In-Context Learning for DocRE

To investigate the effectiveness of LLMs with In-Context Learning (ICL) (Brown et al., 2020; Chowdhery et al., 2022; Wadhwa et al., 2023; Ozyurt et al., 2024) in our task, we compare the LLM-ICL results with the above BERT-based models. Table 9 in Appendix C shows a prompt example we used in our experiments. Specifically, we instruct an LLM to generate a bulleted list of triples for the given passage and the entity list.[5] Each entity is represented in the form "* <ID> : <NAME>" and the entity list is presented as a bulleted list. In the prompt, we also use 3 examples randomly sampled from the training set as the few-shot demonstrations. The same demonstrations are used for all test passages. From each bulleted line generated, we extract the head entity ID e_i, the relation label r, and the tail entity ID e_j using regular expressions. If the extracted entity IDs (e_i, e_j) and the relation label (r) cannot be found in the given entity list $\{e_1, \cdots, e_K\}$ and the possible relation classes $\mathcal{R}$, the bulleted line is ignored. We also remove duplicated triples. The resulting triples are then compared with the gold triples for evaluation.

5 Experiments on Process Identification

Binary classification. In the supervised task formulation, each passage is associated with all 360 process names, with binary labels assigned based on the presence of them in the annotations. Numerous negative samples are constructed for each passage. Consequently, we incorporate negative sampling using various ratios of negative to positive samples. The classification task involves finetuning BERT-based models – BioBERT (Lee et al., 2019), SciBERT (Beltagy et al., 2019), and PubMedBERT (Gu et al., 2021) on the training set. For hyper-parameter details see Table 11.

Generative experiments. Two instruction types are utilized: One prompts the model to list all biological processes in the text, while the other instructs it to generate pairs of processes having a relation. This distinction stems from annotation being conducted at the relation level, where only processes involved in relations are annotated. Consequently, other processes may exist in the text but aren't annotated. We employ Llama2 13B (Touvron et al., 2023) and Llama3 8B (AI@Meta, 2024) on the test set to ensure comparability with the supervised method's results.

5.1 Clustering-based Demonstration Selection

In the zero-shot setting, process names differ significantly from the provided annotations. In the few-shot scenario, performance is highly sensitive to the chosen demonstrations (Li et al., 2022; Lu et al., 2022; Zhang et al., 2023). To enhance few-shot performance, we introduce a retrieval module based on semantic similarity to cluster the most relevant examples from the annotated processes in the development set.

To achieve this, we use the development set to create 10 clusters with K-means clustering (Lloyd, 1982; MacQueen et al., 1967), based on the annotated process list for each passage. Each list is encoded into a vector using BERT by averaging the last hidden state of the [CLS] token for all labels. Each passage in the test set is assigned a cluster given the last hidden state of the [CLS] token of the

[5] In our preliminary experiments, we also tried to generate JSON directly by Llama2 13B; however, generating JSON yielded lower DocRE scores consistently than generating text.

Positive-Negative Ratio	BioBERT			PubMedBERT			SciBERT		
	P	R	F1	P	R	F1	P	R	F1
1:8	21.7	27.9	**24.4**	22.2	42.6	**29.2**	21.3	18.9	20.1
1:4	15.8	45.0	23.4	18.4	60.6	28.2	17.6	44.5	**25.3**
1:1	10.6	64.9	18.3	12.0	47.3	19.2	9.99	71.5	17.5

Table 2: Results of the BERT-based supervised approach on process identification. Precision (P), Recall (R), and F1 scores on the test set of the HOIP dataset are reported. Values in bold represent the best F1 score for each model.

Method	Top-K	In-Ontology Matching			In-Dataset Matching		
		P	R	F1	P	R	F1
Llama2 13B	1	10.7	22.3	14.5	11.2	28.9	16.2
	3	10.0	30.8	15.1	6.7	45.5	11.7
	5	8.3	32.7	13.3	5.4	57.3	9.9
	10	7.0	38.9	11.9	4.1	74.9	7.8
Llama3 8B	1	43.1	11.8	18.6	34.3	28.4	31.1
	3	39.5	16.1	22.9	19.5	43.6	26.9
	5	33.9	19.0	24.3	15.6	55.9	24.4
	10	29.4	27.5	28.4	9.5	62.1	16.5

Table 3: Results of the LLM-based in-context learning approach on process identification. For In-Ontology Matching, we used all entities found in the HOIP ontology as the candidate entities for matching. Matched entities that are not present in the HOIP dataset are ignored. For In-Dataset Matching, we used entities only found in the HOIP dataset as the candidate entities for matching.

passage to the clustering model. During generation, examples from the assigned cluster are included in the prompt. See Table 10 for details.

5.2 Evaluation Methods

We evaluate three primary aspects of generative process identification. The initial aspect involves the Direct Output assessment, where we directly evaluate the output by comparing it to the annotation. This process aims to weight the model's ability to generate processes formulated within the knowledge base framework. As for the second aspect – assessing the system's capability to automatically populate the target knowledge base, we include an ontology alignment-based evaluation which is presented in three steps:

1. Computing embeddings: Let $E_{\text{generated}}$ be the set of embeddings of the generated processes and E_{ontology} be the set of embeddings of ontology terms calculated by SapBERT (Liu et al., 2021).
2. Selecting top k generated processes: For each x_i from $E_{\text{generated}}$, calculate $\text{sim}(x_i, y_j)$ for all y_j from E_{ontology}. Then, select the k elements with the highest cosine similarity.
3. Computing precision, recall and F1 scores between the list of annotated processes and the flattened list of top k generated processes at a passage level.

The last aspect of evaluation mirrors the second one, with the distinction being that instead of matching with the entire ontology, only the process names used in the dataset are taken into account. This approach is grounded on the assumption that the terms utilized in the dataset annotation are the most prevalent.

5.3 Results and Discussion

Direct output assessment. We compared annotation labels with generated process names using zero-shot, regular few-shot, and ICL few-shot settings. The ICL few-shot method achieved the most exact matches, with 30 compared to 2 for regular few-shot and none for zero-shot, underlining the importance of better selected demonstrations, as in Min et al. (2022). Thus, evaluation using the HOIP ontology and dataset matching will be based on the ICL few-shot setting outputs.

In-Dataset Matching. We report in Table 2 the results of the fine-tuned BERT-based models. The negative ratio significantly influences the overall performances of the models. Results indicate that with fewer negative samples, models are more likely to identify true positives but at the cost of also misclassifying more false positives. This is likely due to semantic similarity among the inputs.

PubMedBERT achieves the highest F1 score, under the optimal negative ratio of 8.

Furthermore, we compare the results of the supervised approach with the top-1 results of Llama2 and Llama3 present in Table 3. Across all negative ratios, the F1 score of BERT-based models exceeds the results of Llama2, even under the low-resource setting. However, this trend changes with Llama3, which outperforms the PubMedBERT result by nearly 2 points. Comparing Llama2 and Llama3 reveals that Llama3 is more effective at generating well-tailored process names, resulting in higher precision. Llama3 generates fewer, but better-quality candidates, enhancing task performance. Llama2 improves with more candidates, increasing chances of correct matches, but accuracy still depends significantly on the quality of these generated candidates.

In-Ontology Matching. Following the same tendency in the In-Dataset setting, matching with better-generated process names proves to be more effective overall. Since the goal of this matching is to automatically populate an ontology and DocRE is the next step in the pipeline, concentrating on finding the correct process names is crucial. This focus will aid the DocRE step in serving as a filtering mechanism, ensuring more accurate and relevant candidate triplets to be added to the ontology.

6 Experiments on Document-level RE

We evaluate our systems on the HOIP dataset and CDR dataset (Li et al., 2016). CDR consists of 1,500 abstracts from PubMed, manually annotated with Chemical or Disease entities and Chemical-Induce-Disease relations between them. Since entity IDs in CDR are MeSH unique IDs (e.g., D006493), and the HOIP ontology, while highly specialized for annotating processes related to COVID-19, does not provide the coverage for general chemical compounds and disease terms as the MeSH controlled vocabulary, we used MeSH instead of the HOIP ontology for CDR. We use precision, recall, and F1 metrics, and report the scores averaged independently over 3 trials with different random seeds. A triple (e_i, r, e_j) is considered correct when the head entity ID (e_i), the tail entity ID (e_j), and the relation label (r) are all predicted correctly. We used greedy decoding in the LLM-ICL methods, the results of which do not depend on the seed differences. Table 12 shows the hyperparameters for our DocRE models.

Method	**P**	**R**	**F1**
ATLOP (all mentions)	**64.61**	**75.92**	**69.74**
Llama3 8B (all mentions)	42.26	48.69	45.25
QA-Model (first mention)	56.40	67.39	61.36
MA-ATLOP (first mention)	57.54	**68.11**	**62.34**
MA-ATLOP (first mention) + NES	**57.55**	67.95	62.31
Llama3 8B (first mention)	43.62	49.34	46.30
QA-Model	53.37	64.01	58.12
MA-ATLOP	53.72	65.92	59.18
MA-ATLOP +NES	**54.03**	**66.20**	**59.50**
Llama3 8B	44.75	51.97	48.09

Table 4: DocRE results on the CDR test set. All metrics are averaged over 3 trials. "all mentions" (or "first mention") indicates that the models use all mentions (or the first-appearing mention) as the entity names instead of the canonical name retrieved from the MeSH ontology. The best scores are in bold for each block.

6.1 Experiments on the CDR Dataset

To investigate the importance of mentions in DocRE and how well our models can identify relations without relying on mentions, we first evaluate the models on CDR. We also evaluate a variant of each of our models, which uses the first-appearing annotated mention as the entity name rather than the canonical name retrieved from the MeSH ontology. Although this variant still does not use mention spans, we expect this variant to recognize more easily how the entity is described in the passage than the original model, because the entity names appear at least once in the passage.

Table 4 shows the results. ATLOP exploits mention spans as the direct hints for entity encoding and achieves an F1 score of 69.7. In contrast, our best mention-agnostic model, i.e., MA-ATLOP (+ first mention), achieved an F1 score of 62.3, lower than the ATLOP score by 7.4 points. When there were no mention hints at all, MA-ATLOP and QA-Model yielded F1 scores of 59.2 and 58.1, respectively. These results suggest that our models can identify triples more accurately than expected even without mention hints; however, mentions are still crucial in this task. Also, the BERT-based supervised models outperformed the LLM counterparts. MA-ATLOP outperformed QA-Model consistently. Considering that MA-ATLOP also has higher computation efficiency than QA-Model, MA-ATLOP is more suitable for real-world applications. By employing Negative Entity Sampling (NES), when no mention is available, MA-ATLOP improved all metrics slightly, suggesting the effectiveness

Method	Entity	P	R	F1
QA-Model	gold	51.5	**63.1**	56.7
MA-ATLOP	gold	67.2	52.6	**58.9**
MA-ATLOP + NES	gold	**71.2**	48.6	57.7
Llama3 8B	gold	18.5	16.7	17.6
Upper-bound	pred.	100.0	26.8	42.3
MA-ATLOP	pred.	7.7	14.9	10.2

Table 5: DocRE results on the HOIP test set. The upper and lower blocks show the results when using the ground-truth entities or predicted entities, respectively. The predicted entities are provided by Llama3 8B.

of NES. For the "first-mention" setting, NES did not improve the performance, probably due to the discrepancy between the entity-name style between positive entities (mention) and negative entities (ontology-based name).

6.2 Experiments on the HOIP Dataset

The upper block in Table 5 shows the results on the HOIP dataset when using the ground-truth entities. We evaluate the models that do not require mention hints on this dataset. The BERT-based models achieved much higher F1 scores (56.5-59.0) than LLM (17.6). MA-ATLOP also outperformed QA-Model by 2.2 points in F1. These results were consistent with the results on CDR, demonstrating the effectiveness of MA-ATLOP in terms of both accuracy and computational efficiency in this task. Negative Entity Sampling improved the precision by 4 points, but decreased the recall. These results suggest that, while NES enhances the filtering capability of MA-ATLOP, NES also has the effect of making the model reluctant about positive predictions, and it would be necessary to develop techniques to avoid such biases.

The above experiments assume that the entities are fully and correctly annotated. This setup is appropriate for a clean measurement of the DocRE system's performance itself. However, in reality, entities can be predicted automatically. To evaluate the whole system's performance in the real-world situations, we evaluate our best DocRE model (MA-ATLOP) on the HOIP dataset with entities predicted by Llama3 (8B).

The lower block in Table 5 shows the results. We first calculated the upper-bound scores for the predicted entities. Specifically, we created a subset of gold triples that can be created based on the predicted entities. Precision, recall, and F1 scores are 100.0, 26.8, and 42.3, respectively. The precision does not depend on the quality of predicted entities. The lower recall suggests that there is much room for improvement in DocRE as the process identification recall improves. MA-ATLOP yielded 7.7, 14.9, and 10.2 scores for precision, recall, and F1, respectively. Compared to the much higher precision (67.2) in the gold-entity setup (Table 5), the results suggests that the current model struggles to filter out noisy triples with irrelevant entities. In summary, both improvements in recall (coverage) and precision (low-noisiness) in process identification and DocRE are needed in the current situation, suggesting the difficulty of this task.

7 Case Study

We performed case study to analyze the system outputs qualitatively. We used the best Llama3 (8B) and MA-ATLOP models for process identification and DocRE, respectively.

Table 6 shows an example with true-positive and false-negative entities and triples. Additional example can be found in Appendix F. For ease of understanding, entities are shown by names, not by IDs. We can observe that entities that are almost explicit in the passages, such as “pyroptosis” (*Pyroptosis* in the passage), and “pore formation in membrane of other organism” (*Formation of pores*), were accurately extracted. Triples that are almost explicit based on the context, such as (“pyroptosis” has part, “pore formation in membrane of other organism”) and (“pyroptosis”, has result, “release of DAMP molecules by cell rupture”), were correctly identified by the DocRE system. In contrast, implicit (or knowledge-requiring) entities and triples, such as “binding of pattern recognition receptor to DAMPs”, were not identified by the systems. The entity is derived from the interpretation that *these molecules recruit more immune cells*, which requires background knowledge of immunology: DAMP molecules must bind to receptors recognized by immune cells to recruit immune cells.

The quality evaluation reveals several insights: (1) Detailing causal relationships in elucidating disease mechanisms often necessitates background knowledge not explicitly mentioned in articles. This background knowledge is sometimes added to intermediate causal entities by manual annotation. The BERT-based supervised models and LLMs have difficulty in obtaining such background knowledge and understanding the task from limited

ID72. Passage:
Pyroptosis is a highly inflammatory form of lytic programmed cell death that occurs most frequently upon infection with intracellular pathogens and is likely to form part of the antimicrobial response. Pyroptosis can take place in immune cells and is also reported to occur in keratinocytes and some epithelial cells. Formation of pores causes cell membrane rupture and release of cytokines, as well as various damage-associated molecular pattern (DAMP) molecules such as HMGB-1, ATP and DNA, out of the cell. These molecules recruit more immune cells and further perpetuate the inflammatory cascade in the tissue.

True-Positive Entities:
* pore formation in membrane of other organism
* pyroptosis
* release of DAMP molecules by cell rupture

False-Negative Entities:
* binding of pattern recognition receptor to DAMPs

True-Positive Triples:
* (pyroptosis, has part, pore formation in membrane of other organism)
* (pyroptosis, has result, release of DAMP molecules by cell rupture)

False-Negative Triples:
* (release of DAMP molecules by cell rupture, has result, binding of pattern recognition receptor to DAMPs)

Table 6: A case study on our process identification and DocRE models. Phrases referred in Section 7 are underlined.

supervision (labeled data or demonstrations). (2) NLP systems could be complementary to manual annotations. Manual annotation often focuses on the causality of a particular process in one literature and gives priority to further causes and consequences in other literature. Therefore, other processes and causal relationships in the same passage may not be extracted. It is also possible that processes that missed identification due to simple errors due to annotation fatigue are also in the false positive. In such manual annotation issues, NLP analysis could make a significant contribution to the identification of processes.

8 Related Work

Gene Ontology Causal Activity Modeling (GO-CAM) defines molecular-level causal relationships (Thomas et al., 2019); however, it lacks granularity and context for COVID-19 infection. Our HOIP dataset is based on the HOIP ontology (Yamagata et al., 2021, 2024), which organizes knowledge about biomedical processes in the context of COVID-19 infectious courses and thus essential for analyzing SARS-CoV-2 infection and progression.

In knowledge acquisition, entities are typically identified by Named Entity Recognition (NER) and Entity Disambiguation (ED). The task of NER is to identify mentions in the given text that represent one of the pre-defined types (e.g., Chemical, Disease) (Yu et al., 2020; Zhu and Li, 2022; Ye et al., 2022). The entity mentions are then passed to ED to link them to the knowledge-base concept IDs that the mentions refer to best (Kolitsas et al., 2018; Wu et al., 2020; Cao et al., 2021; Yamada et al., 2022). These tasks commonly assume that entities are explicitly described in text. In reality, however, entities are not necessarily explicitly described. In this work, we explore mention-agnostic methods for process identification.

The most widely used approach to DocRE is to model entities by a pre-trained Transformer and perform pairwise relation classification. Christopoulou et al. (2019) proposed to model entity dependencies via graphs with nodes of various granularities. Zhou et al. (2021) proposed ATLOP, which models entity-pair contexts for pairwise relation classification. Xiao et al. (2022) introduced evidence modeling for improving ATLOP. Zhang et al. (2021) used U-Net architecture for modeling entity dependencies. These methods commonly rely on mentions and often insert special mention-boundary markers into text to indicate the mention locations to the Transformer. However, Li et al. (2023) showed that these methods are too sensitive to the accuracy of mentions and it is unrealistic to expect perfect mentions in the real-world scenario. In contrast, we propose mention-agnostic DocRE methods and investigate how well the mention-agnostic models can identify relations.

9 Conclusion

To assist ontology-based biological knowledge annotation, this work proposes a new dataset and practicable entity- and relation-level biomedical information extraction methods. We will continue to promote relevant research of semi-automatic annotation and advance practical applications.

Limitations

Despite demonstrating promising outcomes in mention-agnostic process identification and DocRE, our methodology does face limitations. First, our two-stage IE system consists of a cascade of process identification and DocRE, which inevitably suffers from error propagation. The experimental results in the pipeline setting suggest that the DocRE performance is significantly vulnerable to the accuracy of predicted entities. Moreover, the process identification model and the DocRE model are disconnected and cannot interact with each other. Second, our methods have only been evaluated in the domain of the HOIP ontology, and the accuracy in other biomedical domains and ontologies remains unknown. Third, our methodology has not been fully evaluated by domain experts. Although an expert analysis is performed, the analysis is based primarily on just two examples. A more thorough and detailed analysis by specialists is needed. Tackling these limitations remains an intriguing avenue for future research.

Acknowledgments

We would like to thank the anonymous reviewers for their thoughtful and insightful comments, which we found very helpful in improving the paper. This work was supported by JSPS KAKENHI Grant Numbers JP22K17959, 21K17815, and JP22H05015, and ANR AIBy4 project (ANR-20-THIA-0011), Nantes University.

References

AI@Meta. 2024. Llama 3 model card.

M. Ashburner et al. 2000. Gene ontology: tool for the unification of biology. the gene ontology consortium. *Nat Genet*, 25:25–29.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. *Preprint*, arXiv:2010.00904.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways. *Preprint*, arXiv:2204.02311.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4925–4936.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare*, 3(1).

Nikolaos Kolitsas, Octavian-Eugen Ganea, and Thomas Hofmann. 2018. End-to-end neural entity linking. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 519–529, Brussels, Belgium. Association for Computational Linguistics.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. 2016. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. *Database : the journal of biological databases and curation*.

Jing Li, Yequan Wang, Shuai Zhang, and Min Zhang. 2023. Rethinking document-level relation extraction: A reality check. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5715–5730, Toronto, Canada. Association for Computational Linguistics.

Tianyi Li, Wenyu Huang, Nikos Papasarantopoulos, Pavlos Vougiouklis, and Jeff Z. Pan. 2022. Task-specific pre-training and prompt decomposition for knowledge graph population with language models. *ArXiv*, abs/2208.12539.

Fangyu Liu, Ehsan Shareghi, Zaiqiao Meng, Marco Basaldella, and Nigel Collier. 2021. Self-alignment pretraining for biomedical entity representations. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4228–4238.

Stuart Lloyd. 1982. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

James MacQueen et al. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *Preprint*, arXiv:2202.12837.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, Yejin Choi, and Hannaneh Hajishirzi. 2022. Reframing instructional prompts to GPTk's language. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 589–612, Dublin, Ireland. Association for Computational Linguistics.

Yilmazcan Ozyurt, Stefan Feuerriegel, and Ce Zhang. 2024. Document-level in-context few-shot relation extraction via pre-trained language models. *Preprint*, arXiv:2310.11085.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2023. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. *Preprint*, arXiv:2310.11324.

Yun Tang, Jing Huang, Guangtao Wang, Xiaodong He, and Bowen Zhou. 2020. Orthogonal relation transforms with graph context modeling for knowledge graph embedding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2713–2722, Online. Association for Computational Linguistics.

P.D. Thomas, D.P. Hill, H. Mi, et al. 2019. Gene ontology causal activity modeling (go-cam) moves beyond go annotations to structured descriptions of biological functions and systems. *Nat Genet*, 51:1429–1433.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Somin Wadhwa, Silvio Amir, and Byron Wallace. 2023. Revisiting relation extraction in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15566–15589, Toronto, Canada. Association for Computational Linguistics.

Qinyong Wang, Zhenxiang Gao, and Rong Xu. 2023. Exploring the in-context learning ability of large language model for biomedical concept linking. *Preprint*, arXiv:2307.01137.

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online. Association for Computational Linguistics.

Yuxin Xiao, Zecheng Zhang, Yuning Mao, Carl Yang, and Jiawei Han. 2022. SAIS: Supervising and augmenting intermediate steps for document-level relation extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2395–2409, Seattle, United States. Association for Computational Linguistics.

Ikuya Yamada, Koki Washio, Hiroyuki Shindo, and Yuji Matsumoto. 2022. Global entity disambiguation with BERT. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3264–3271, Seattle, United States. Association for Computational Linguistics.

Yuki Yamagata, Tsubasa Fukuyama, Shuichi Onami, and Hiroshi Masuya. 2024. Prototyping an ontological framework for cellular senescence mechanisms: A homeostasis imbalance perspective. *Sci Data*, 11:485.

Yuki Yamagata, T. Kushida, Shuichi Onami, and Hiroshi Masuya. 2021. Ontology development for building a knowledge base in the life science and structuring knowledge for elucidating the covid-19 mechanism. In *Proceedings of the Annual Conference of JSAI*, pages 3H1GS3d01–03H1GS03d01.

Deming Ye, Yankai Lin, Peng Li, and Maosong Sun. 2022. Packed levitated marker for entity and relation extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4904–4917, Dublin, Ireland. Association for Computational Linguistics.

Juntao Yu, Bernd Bohnet, and Massimo Poesio. 2020. Named entity recognition as dependency parsing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6470–6476, Online. Association for Computational Linguistics.

Biao Zhang, Barry Haddow, and Alexandra Birch. 2023. Prompting large language model for machine translation: A case study. *Preprint*, arXiv:2301.07069.

Ningyu Zhang, Xiang Chen, Xin Xie, Shumin Deng, Chuanqi Tan, Mosha Chen, Fei Huang, Luo Si, and Huajun Chen. 2021. Document-level relation extraction as semantic segmentation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 3999–4006. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Heliang Zheng, Jianlong Fu, Zheng-Jun Zha, and Jiebo Luo. 2019. Learning deep bilinear transformation for fine-grained image representation. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. 2021. Document-level relation extraction with adaptive thresholding and localized context pooling. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 14612–14620.

Enwei Zhu and Jinpeng Li. 2022. Boundary smoothing for named entity recognition. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7096–7108, Dublin, Ireland. Association for Computational Linguistics.

A HoIP Ontology

Understanding the HOIP ontology may be helpful for understanding our HOIP dataset and the task. In this section, we describe the features of the HOIP ontology.

The HOIP ontology is annotated based on COVID-19 related articles in PubMed using Protégé 5.5.0[6] and the Web Ontology Language (OWL). The COVID-19 infectious processes are manually annotated. Passages corresponding to the annotated terms are also provided. Article identifiers (e.g., PubMed ID (PMID: 25301932), DOI) are also provided using the database cross-reference annotation property.

The processes in HOIP consist of a hierarchy. The infectious processes described in the articles and the superclass of each process using Gene Ontology are annotated.

The relationships between processes are annotated using object properties, Causal relationships between processes are primarily annotated using the 'has result' relationship. Furthermore, subprocesses of a process are identified using the 'has part' relation.

HOIP defines a "COVID-19 infectious course" as a sequence of the abovementioned processes to describe infectious mechanisms. These courses are organized into an is-a (subclass of) hierarchy by severity, ranging from mild to severe. Notably, the "COVID-19 severe course" includes a subclass associated with acute respiratory distress syndrome (ARDS). These COVID-19-specific processes are used as our primary dataset for this study.

B Question Templates

Table 7 shows the question templates $\mathcal{T}_r$ $(r \in \mathcal{R})$ used for QA-Model. In the table, <HEAD> and <TAIL> are replaced by the head and tail entity names, respectively. The entity names are retrieved from the ontology using the entity IDs as query. We

[6] `https://protege.stanford.edu`

Dataset	Relation	Question Template
CDR	CID	*Does <HEAD> induce <TAIL> ?*
HOIP	has result	Does <HEAD> result in <TAIL> ?
	has part	*Does <HEAD> involve <TAIL> ?*
	has molecular reaction	*Does <HEAD> have molecular reaction of <TAIL> ?*
	part of	*Is <HEAD> part of <TAIL> ?*

Table 7: Question templates used in QA-Model (Section 4.1).

manually created the question templates for each dataset: CDR and HOIP.

C Prompts

Table 8 shows an example of the prompt used in the few-shot setting in process identification. Only examples section is discarded in the zero-shot setting. Table 9 also shows a prompt used in DocRE experiments on the HOIP dataset and the corresponding output by Llama3 (8B). We replaced the ontology name ("HOIP") and possible relation classes ("has-result, has-part, ...") in the prompt template with "MeSH" and "Chemical-Induce-Disease" respectively in CDR experiments. The demonstrations are also different between the datasets.

D ICL Few-Shot Setting in Process Identification

Table 10 exhibits the number of examples per cluster created for ICL in the few-shot setting in process identification.

E Hyperparameters

Table 11 shows the hyper-parameters used in the supervised models for process identification. Table 12 also list hyper-parameters used in our DocRE models.

F Another Example of Case Study

Table 13 shows another example used in our case study (in Section 7).

Instruction:	Generate the list of processes present in the Text.
Constraints:	Don't repeat the question. Justification and explanation are prohibited.
Examples:	**Text:** Within 19 days after symptom onset, 100% of patients tested positive for antiviral immunoglobulin-G (IgG). Seroconversion for IgG and IgM occurred simultaneously or sequentially. **Answer:** [immunoglobulin production, immunoglobulin mediated immune response]
Text:	ACE2 expression has been demonstrated in arterial and venous endothelium of several organs, and histopathological studies have found microscopic evidence of SARS-CoV-2 viral particles in endothelial cells of the kidneys and lungs.
Answer:	-

Table 8: Example of the few-shot setting prompt in process identification, following the described prompt template.

Prompt:

Based on the given text and entities associated with the text, please identify relations between the entities.
1. Named entities are listed next to the text.
2. Each entity is represented using HOIP Concept ID.
3. Possible relations: has-result, has-part, has-molecular-reaction, part-of
4. Output a bulleted list of triples. Each bullet line corresponds to each triple: “<BULLET> (<SUBJECT ENTITY>, <RELATION>, <OBJECT ENTITY>)“,
where <SUBJECT ENTITY>, <RELATION>, and <OBJECT ENTITY>, correspond to the subject entity, the relation label, and the object entity, respectively.

Below are some examples.

# Example 1
Text: We also provide biophysical and structural evidence that ...
Entities:
 * http://purl.bioontology.org/ontology/HOIP/HOIP_0040511: Negative regulation of ACE2 activation
 * http://purl.bioontology.org/ontology/HOIP/HOIP_0041139: host cell surface receptor binding in nasal epithelial cells
...
Answer:
 1. (http://purl.obolibrary.org/obo/GO_0046789, has-result, http://purl.bioontology.org/ontology/HOIP/HOIP_0040511)
 2. (http://purl.obolibrary.org/obo/GO_0046789, has-part, http://purl.bioontology.org/ontology/HOIP/HOIP_0041139)
...

# Example 2
...
# Example 3
...

Let’s try the following test example.

# Test Example
Text: Within 19 days after symptom onset, 100% of patients tested positive for antiviral immunoglobulin-G (IgG). Seroconversion for IgG and IgM occurred simultaneously or sequentially.
Entities:
 * http://purl.obolibrary.org/obo/GO_0002377: immunoglobulin production
 * http://purl.obolibrary.org/obo/GO_0016064: immunoglobulin mediated immune response

Please output the answer to the test example in bullet points, following the format specified above.

Generated response:

Here is the answer to the test example:

• (http://purl.obolibrary.org/obo/GO_0002377, has-result, http://purl.obolibrary.org/obo/GO_0016064)

Note that there is only one possible relation between the entities in this example, which is "has-result".

Table 9: Example of the whole prompt in DocRE and the corresponding Llama3 (8B) output.

Cluster	0	1	2	3	4	5	6	7	8	9
# Elements	3	9	4	4	1	5	3	1	4	1

Table 10: Number of elements for each cluster created for ICL few-shot setting. In cluster assignement, only clusters 0, 2, 3, 5, 6 appear in the HOIP test set

Hyper-parameter	
Max Sequence Length	512
Optimizer	AdamW
Learning Rate	1×10^{-5}
Weight Decay	1×10^{-6}
Epochs	8
Batch Size	8

Table 11: Hyper-parameters for the supervised models on Process Identification.

Hyper-parameter	**QA-Model**	**MA-ATLOP**	**LLM-ICL**
Pre-trained model	SciBERT (cased)	SciBERT (cased)	Llama3 (8B; instruction-fine-tuned)
Max Sequence Length	512	512	4096
Bilinear Group G	-	64	-
Negative Sampling Ratio ρ	-	0.5	-
Optimizer	AdamW	AdamW	-
Learning Rate (BERT encoders)	2×10^{-5}	2×10^{-5}	-
Learning Rate (FFNNs)	1×10^{-4}	1×10^{-4}	-
Epochs	30	30	-
Batch Size	4	2	-
Warmup Ratio	0.06	0.06	-
# Few-Shot Examples	-	-	3
Quantization Bits	-	-	4
dtype	-	-	BFloat16
Max. New Tokens	-	-	512

Table 12: Major hyper-parameters for the DocRE models.

ID242. Passage:

Moreover, isolated right ventricular dysfunction may occur as a result of elevated pulmonary vascular pressures secondary to ARDS, pulmonary thromboembolism, or potentially virus-mediated injury to vascular endothelial and smoothmuscle tissue.

True-Positive Entities:

* increasing blood pressure
* respiratory blood vessel smooth muscle damage
* thrombus formation

False-Negative Entities:

* artery narrowing
* endothelium damage
* endothelium malfunction
* vasoconstriction

True-Positive Triples:

* (endothelium damage, has result, endothelium malfunction)
* (thrombus formation, has result, artery narrowing)
* (vasoconstriction, has result, increasing blood pressure)

False-Negative Triples:

* (respiratory blood vessel smooth muscle damage, has result, vasoconstriction)

Table 13: A case study on our process identification and DocRE models.

Automatic Extraction of Disease Risk Factors from Medical Publications

Maxim Rubchinsky[1] Ella Rabinovich[1] Adi Shraibman[1] Netanel Golan[3,4]
Tali Sahar[4] Dorit Shweiki[2]

[1]School of Computer Science, The Academic College of Tel Aviv-Yaffo, Israel
[2]Bioinformatics, School of Computer Science, The Academic College of Tel Aviv-Yaffo, Israel
[3]Division of Cardiology, Tel-Aviv Sourasky Medical Center, Tel Aviv, Israel
[4]Faculty of Medicine, The Hebrew University of Jerusalem, Israel

maxim@rubchinsky.com, {ellara,adish,dorits}@mta.ac.il, Netanel.golan@mail.huji.ac.il

Abstract

We present a novel approach to automating the identification of risk factors for diseases from medical literature, leveraging pre-trained models in the bio-medical domain, while tuning them for the specific task. Faced with the challenges of the diverse and unstructured nature of medical articles, our study introduces a multi-step system to first identify relevant articles, then classify them based on the presence of risk factor discussions and, finally, extract specific risk factor information for a disease through a question-answering model.

Our contributions include the development of a comprehensive pipeline for the automated extraction of risk factors and the compilation of several datasets, which can serve as valuable resources for further research in this area. These datasets encompass a wide range of diseases, as well as their associated risk factors, meticulously identified and validated through a fine-grained evaluation scheme. We conducted both automatic and thorough manual evaluation, demonstrating encouraging results. We also highlight the importance of improving models and expanding dataset comprehensiveness to keep pace with the rapidly evolving field of medical research.

1 Introduction

Automatic identification of risk factors for diseases plays a pivotal role in preventive medicine, enabling healthcare professionals to formulate effective prevention strategies and improve patient outcomes. Traditionally, this process has relied heavily on manual review of extensive medical literature, a time-consuming and labor-intensive task, hindering knowledge accessibility and effective usage.

As a concrete example, recently, compelling evidence has emerged linking Lipoprotein A (Lp(a)) — a particle operating similarly to the more familiar LDL molecule — to the pathogenesis of atherosclerosis and subsequent coronary artery disease, commonly referred to as Myocardial Infarction (MI). Despite the established role of Lp(a) as a risk factor (Kronenberg et al., 2022), many primary care clinicians remain inadequately informed, occasionally lacking knowledge regarding its testing procedures. Moreover, in a conversation with a board-certified professor of interventional cardiology, he disclosed receiving frequent inquiries from other clinicians questioning the necessity of referrals for Lp(a) testing. This highlights the pressing need for an automated tool capable of screening vast amounts of scientific literature and identifying prominent risk factors for various diseases.

Despite significant advances in the field of natural language processing, automatic extraction of disease risk factors from *scientific medical literature* remains a challenging endeavor. Contrary to the analysis of *electronic health records* (Chen et al., 2015; Boytcheva et al., 2017; Chokwijitkul et al., 2018), here the primary challenge lies in the diverse and unstructured nature of medical publications, where risk factors are described in various contexts and formats. What is more, the continuous discovery of new risk factors necessitates a dynamic approach that can adapt to the evolving body of medical knowledge. This study introduces a novel approach to automating the identification of disease risk factors from medical literature.

Utilizing pre-trained large language models, based on BioBERT (Lee et al., 2020), we developed a multi-step system, that first identifies relevant medical articles, classifies them based on the presence of risk factor discussions, and then extracts specific risk factor information through a question answering (QA) model. Our approach to extraction of disease risk factors is illustrated in Figure 1: (1) medical abstracts are retrieved from PubMed, (2) a specifically fine-tuned binary classifier is used to identify abstracts with risk factors information, and (3) textual spans containing risk factors are extracted via a question answering model, fine-tuned on manually annotated QA items.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 474–485
August 16, 2024. ©2024 Association for Computational Linguistics

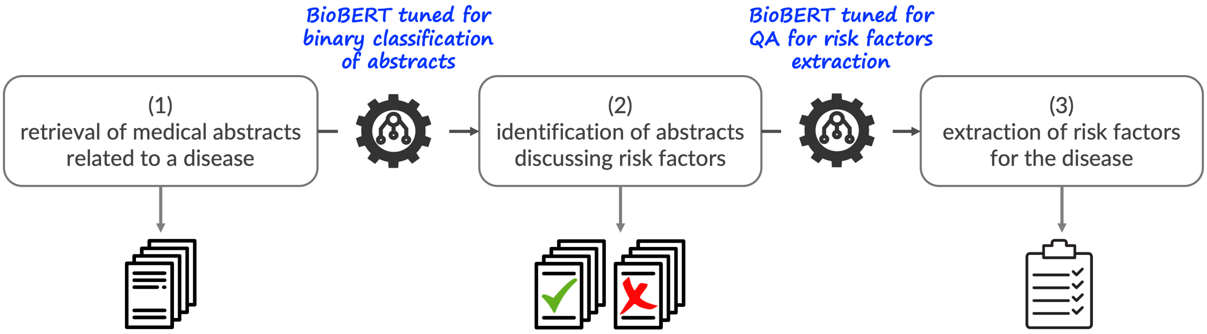

BioBERT tuned for binary classification of abstracts
BioBERT tuned for QA for risk factors extraction
(1) retrieval of medical abstracts related to a disease
(2) identification of abstracts discussing risk factors
(3) extraction of risk factors for the disease

Figure 1: The pipeline for extraction of disease's risk factors: (1) medical abstracts are retrieved from PubMed, (2) a specifically fine-tuned binary classifier is used to identify abstracts with risk factors information, and (3) precise textual spans containing risk factors are extracted via a QA model, fine-tuned on manually annotated QA items.

The contribution of this work, therefore, twofold: First, we present a comprehensive pipeline for automated extraction of risk factors. Second, we compile and make available several datasets that can serve as valuable resources for future research in this field. These datasets include a carefully annotated, large and diverse set of over 1,700 risk factors associated with 15 diseases, as well as set of over 160,000 automatically extracted risk factors,[1] with almost 1,500 manually assessed for their quality, using a fine-grained annotation scheme.[2]

We survey the related work in Section 2 and detail on collection and annotation of our datasets in Section 3. We next describe our approach to the task and report experimental results in Section 4. Human evaluation results are presented in Section 5. Discussion of the difficulty of the task and the limitations of this work are presented in Section 6. We conclude this study in Section 7.

2 Related Work

Automatic identification of disease risk factors through the analysis of medical texts has garnered interest across various research domains, particularly in applying natural language processing and machine learning techniques to electronic health records (EHRs) and electronic medical records (EMRs). Here we review key contributions in this area, highlighting approaches that parallel and diverge from our focus on free-text medical articles.

Chokwijitkul et al. (2018) explore the utilization of deep learning models to extract heart disease risk factors from EHRs. The approach, grounded in analyzing structured data within EHRs, contrasts with our exploration of unstructured text in medical literature, underscoring the diversity in data sources for risk factor identification. Boytcheva et al. (2017) attempt at mining clinical texts for risk factor identification using association rules. Specifically, they handle data in XML format from the Diabetes Register, indicating a structured approach to data analysis. This work differs from ours in terms of both data source type (clinical narratives), as well as in our broader application to unstructured, free-text medical articles and the use of pre-trained large language models (LLMs) for the task of text understanding.

A comprehensive work on identifying risk factors for heart disease (from clinical data) over time was done in a shared task organized by UTHealth[3] (Stubbs et al., 2015). Sheikhalishahi et al. (2019) offer an overview of NLP applications in analyzing clinical notes for chronic disease management, highlighting the increasingly significant contribution of language models to healthcare applications. In the domain of precision medicine, Sabra et al. (2017) focus on extracting semantic information and assessing sentiments in clinical notes.

Various works have employed data mining and machine learning (ML) techniques for identifying risk factors from patient data (Abdelhamid et al., 2023), or clinical outcome prediction (Kavakiotis et al., 2017; Mehmood et al., 2021; Naik et al., 2021). Recently, the identification of risk factors for delirium prediction, a rare adverse reaction observed in COVID-19 patients, was developed utilizing ML applied to nursing records (Miyazawa et al., 2024). Additional line of studies focuses

[1] We note that the set of over 160,000 automatically extracted risk factors are of admittedly mixed quality (see Section 5 and Table 5 for details), yet, we thought this data can serve the community for further research in the field.

[2] All code and data are available at `https://github.com/maximrub/diseases-risk-factors`.

[3] The University of Texas Health Science Center.

on building language models specifically-tailored for medical literature related tasks (Roitero et al., 2021; Yang et al., 2022; Singhal et al., 2023).

Several significant contributions have been made in the field of biomedical relation extraction, which includes identifying factors that predispose individuals to diseases. The SemRep (Kilicoglu et al., 2020) tool extracts semantic predications from biomedical texts, including relationships such as "predisposes". The outputs of SemRep have been used to create SemMedDB (Kilicoglu et al., 2012), a large-scale repository of semantic predications from PubMed. Building on these resources, BioPREP (Hong et al., 2021) employs deep learning techniques for predicate classification. The BioRED (Luo et al., 2022) dataset includes a "positive correlation" relation between diseases and other biomedical entities like genes and chemicals.

Conclusion While the majority of existing research focuses on analyzing structured *electronic health records* and *electronic medical records* to identify disease risk factors, our study pushes beyond these confines by examining free-text medical literature. Processing unstructured medical text introduces distinct challenges, especially due to language complexity, variation, and the potential for nuanced double meanings, and even worse, due to the necessity to discern context accurately. Consequently, it opens up expansive opportunities for subtle understandings of disease risk factors, facilitating both research and practical applications.

3 Dataset

Data collection process for this work can be viewed as a three-step process: (i) collection of the set of disease names spanning multiple disease families, (ii) manual annotation of scientific article abstracts containing explicit mention of risk factors of a subset of diseases – "abstracts seed", and (iii) manual annotation of risk factors description (span) in abstract texts found in (ii) – "risk factors seed". We detail on each step in this multi-phase procedure.

3.1 Disease Dataset Collection

Aiming to assemble a comprehensive list of diseases, we made use of the KEGG Disease Database API[4] to retrieve disease-related information, including names, description and relevant medical codes such as MeSH (Medical Subject Headings), ICD-10 and ICD-11.[5] This process resulted in 2,624 distinct disease names, comprising the foundation for further retrieval of scientific abstracts and, ultimately, automatic extraction of risk factors, from scientific medical literature.

3.2 Seed Dataset with Relevant Abstracts

Retrieval of Abstracts Discussing Risks Using the list of disease names retrieved from KEGG, we next queried PubMed[6] — a large, reliable, and authoritative resource of biomedical literature — for article abstracts containing the disease names. Specifically, we used the Entrez Programming Utilities[7] via the biogo package.[8] The inherent limitation of this study is related to the fact that only abstracts are freely available through the PubMed interface. However, paper abstracts typically contain a concise summary and main findings of the work, hence constitute a sufficient input for the task at hand. Similarly, prior studies analyzed abstracts retrieved from PubMed for building a biological network (Chen and Sharp, 2004), topical clustering (David and Samuel, 2012), and identification of negative and positive domain-specific medical terms (Vinkers et al., 2015).

Aiming at retrieval of abstracts discussing findings related to risk factors, we queried PubMed for containment of the phrase "risk factor" in a paper's information: title, abstract or MeSH terms. The following pseudo-code was used for this purpose:

```
select articles from datastore where

[disease_name] in {title|abstract|MeSH_terms}
and "risk factor" in {title|abstract|MeSH_terms}
```

where disease_name refers to the disease we are seeking risk factors for, and the exact search term "risk factor" (surfacing also the plural "risk factors") can appear in abstract, title or MeSH terms.

Annotation of Abstracts for Risk Factors Despite the evident potential, not every abstract with explicit mention of "risk factor" or marked with a "risk factor" MeSH term contains risk factors for a pre-defined disease. As a concrete example, a medical study can mention a list of potential risk factors tested, without any of them showing as significant.

[4] KEGG database: https://www.kegg.jp/kegg/disease/; specifically, we used its REST API service at https://www.kegg.jp/kegg/rest/ for retrieval.

[5] As of April 2024, ICD-11 (International Classification of Diseases, v11) is the most up-to-date code collection.

[6] https://pubmed.ncbi.nlm.nih.gov

[7] https://www.ncbi.nlm.nih.gov/books/NBK25501

[8] https://github.com/biogo/ncbi

We, therefore, define our first (pre-processing) task as automatic classification of a retrieved abstract for spelling out an artifact, found to be a risk factor for the disease in the study.

A qualified annotator with medical background (one of the authors of this paper) annotated a random set of 182 abstracts. The procedure resulted in 87 positive abstracts (explicitly mentioning a risk factor) and 95 negative, thereby comprising a sufficient training set for the binary classifier – step (2) in the pipeline in Figure 1. Table 1 shows two examples of relevant abstract parts containing risk-related phrases which do or do not qualify as risk factors, as identified by the annotator. Evidently, the nuanced language used to discuss risks in various contexts renders the task as non-trivial for both humans and automatic tools.

3.3 QA Seed Dataset with Risk Factors

Given an article abstract specifying a risk factor(s) for a certain disease, we cast the risk factor identification problem as *extractive question answering* scenario, where given the abstract and the question `"What are the risk factors for {disease name}?"`, a textual span, containing the answer, will be identified. In Section 4.1.2 we make use of the established and popular BERT-based QA model – BioBERT[9] (Lee et al., 2020), and fine-tune it for the task at hand using a manually annotated set of QA items: context (article abstract), a targeted question of the form mentioned above, and a set of manually marked answers in the form `span_start` and `answer_text` (implying `span_end`).

In the absence of suitable annotated datasets for this nuanced task, we developed a web interface for medical students to manually annotate article abstracts. This interface is used for (manual) identification of text segments within abstracts, given the disease discussed in the article. We present a few screenshots of the annotating tool in Appendix A.1, and release the tool for the community.

The annotator with medical background marked text spans containing risk factors in a random set of 668 abstracts identified to contain explicit mention of a risk factor,[10]. resulting in the total of 1,712 QA items, spanning 15 diverse diseases,[11] where each QA item reflects a single risk factor in an abstract that (possibly) encompasses multiple valid risks. Sentences suggesting risk factors significant only within specific population subgroups were denoted as such. Table 4 presents two examples of QA items: disease name, abstract, and the highlighted risk factor span, as marked by the annotator.

Collectively the carefully-curated and annotated set of abstracts for binary classification of medical articles, and the set of QA items, comprise a high-quality collection for tuning pre-trained language models for the purpose of this study.

4 Methodology and Experiments

We further describe in detail our methodological approach, experimental setup and results.

4.1 Methodology

As illustrated in Figure 1, we apply a multi-step approach to automate the identification of disease risk factors from medical literature. Central to our methodology is the use of BioBERT, a variant of BERT pre-trained on biomedical texts, enabling nuanced understanding of complex medical language (Lee et al., 2020). We next provide details on each step in the process. This model was chosen due to its proven benefits in the biological domain, and its encoder-based architecture – (arguably) the most appropriate choice for both the classification and extractive question answering tasks at hand.[12]

4.1.1 Detection of Abstracts with Risk Factors

The pre-trained BioBERT-based classifier[13] was tuned for abstracts classification using the training part (80%) out of over 182 manually annotated abstracts (see Section 3.2), and tested on the held-out part (20%), achieving the accuracy of 92%. Table 3 reports the per-class classification results. This encouraging result facilitated our efforts of analyzing content that is most likely to yield valuable insights into disease-risk factor associations.

We collected a substantial dataset of abstracts, by querying PubMed for each one of over 2400 diseases, as detailed in Section 3.2; this step resulted in 137,740 abstracts. We next apply the fine-tuned classifier to identify abstract potentially containing risk factors for a disease. Out of the total number of 137,740 abstracts, 89,834 were classified as positive – containing explicit mentions of risk

[9] `https://huggingface.co/dmis-lab/biobert-v1.1`

[10] The abstracts were sampled from the set automatically classified as "positive" (see Section 4.1.1)

[11] Appendix B reports the full list of diseases.

[12] Our future work includes investigation of decoder-based models (e.g., GPT), casting the QA part as an abstractive task.

[13] `https://huggingface.co/dmis-lab/biobert-v1.1`. We used the default settings with max_input_length of 512 tokens, training the classifier for three epochs.

article title: **Risk Factors for Pediatric Human Immunodeficiency Virus-related Malignancy** (2003)

Context: Although cancers occur with increased frequency in children with human immunodeficiency virus (HIV) infection, the specific clinical, immunological, and viral risk factors for malignancy have not been identified. **Objective:** To identify risk factors for malignancy among HIV-infected children. [...] Epstein-Barr virus viral load of more than 50 viral genome copies per 105 peripheral blood mononuclear cells was strongly associated with cancer risk but only for children with CD4 cell counts of at least 200/ microL (odds ratio [OR], 11.33; 95% confidence interval [CI], 2.09-65.66, P<.001). [...]
High viral burden with EBV was associated with the development of malignancy in HIV-infected children although the effect was modified by CD4 cell count. The pathogenesis of HIV-related pediatric malignancies remains unclear and other contributing risk factors can be elucidated only through further study.

article title: **Profound Hypoglycemia and High Anion Gap Metabolic Acidosis in a Pediatric Leukemic Patient Receiving 6-Mercaptopurine** (2024)

A 13-year-old male undergoing maintenance chemotherapy with methotrexate and 6-mercaptopurine (6MP), for very high-risk B-cell acute lymphoblastic leukemia (ALL), presented with vomiting due to severe hypoglycemia with metabolic acidosis. While his laboratory values were concerning for a critically ill child, the patient was relatively well appearing. Hypoglycemia is a rare but serious side effect of 6MP with an unexpectedly variable presentation; therefore, a high index of suspicion is needed for its prompt detection and treatment. [...]
6MP-induced hypoglycemia can be ameliorated with the addition of allopurinol to shunt metabolism in favor of the production of therapeutic metabolites over hepatotoxic metabolites. Additionally, a morning administration of 6MP and frequent snacks may also help to prevent hypoglycemia. Overall, this case adds to the literature of unusual reactions to 6MP including hypoglycemia in an older child without traditional risk factors.

Table 1: Article abstracts discussing risk factors (retrieved per the query in Section 3.2). Top – abstract identified as relevant for risk factors extraction by the annotator, where the highlighted part refers to the discussed factor. Bottom – abstract mentioning "risk factors", yet annotated as irrelevant.

factors for diseases. Naturally, some diseases (and disease families) resulted in more prolific retrieval, due to their higher coverage in the medical literature: while various cancer types (e.g., Carcinoma, Leukemia) have large body of related articles, genetic disorders are surveyed less frequently in the context of risk factor discussion.

4.1.2 Identification of Disease Risk Factors

The collection of abstracts classified positively to contain a risk factor, was then subject to the task of risk factor extraction – step (3) in Figure 1. We cast the task as extractive QA, where the medical abstract represents the context, and the question template is formulated as "`What are the risk factors for {disease name}?`". We anticipate the BioBERT QA model (Lee et al., 2020) to identify span(s) in the abstract containing the answer (or answers, in case multiple risk factors are mentioned in the same abstract), similarly to examples presented in Table 2. We fine-tune the model for the specific task, as described below.

Fine-tuning the QA Model We tuned the BioBERT model for our usecase using the training part (80%) of the 1,712 QA items annotated manually by the author with medical background (see Section 3.3); the remaining 20% were used for testing. Notably, the set of 15 diseases in the 668 abstracts was carefully split into training and test sets, so that the same disease does not appear in both sets, facilitating the assessment of the model's generalizability and performance across a variety of disease contexts. The model tuning was done using the maximum context length of 384 tokens, learning rate of 2e-5, and 25 epochs.

We use two common metrics for automatic evaluation of extractive question answering: `exact-match` and `F1-score`. Applied on the test set (342 QA items), the metrics obtained 61.76% for `exact-match`, and 88.23% for `F1-score`, highlighting the potential of the approach.

Determining the Maximum Answer Length We determined the maximum length for answers in our QA model by analyzing the lengths of all answers within our training dataset. We calculated the length of each answer (in characters) and studied their distribution. The maximum answer length was set at the 95th percentile of these lengths to encompass the majority of real-world answers while excluding outliers. This threshold is crucial for maintaining focus on concise and relevant answer segments, thereby enhancing the model's training

disease: **Diabetes in Men**
OBJECTIVE: To examine the association between smoking, alcohol consumption, and the incidence of non-insulin dependent diabetes mellitus in men of middle years and older. [...] RESULTS: During 230,769 person years of follow up 509 men were newly diagnosed with diabetes. After controlling for known risk factors men who smoked 25 or more cigarettes daily had a relative risk of diabetes of 1.94 (95% confidence interval 1.25 to 3.03) compared with non-smokers. Men who consumed higher amounts of alcohol had a reduced risk of diabetes (P for trend < 0.001). Compared with abstainers men who drank 30.0-49.9 g of alcohol daily had a relative risk of diabetes of 0.61 (95% confidence interval 0.44 to 0.91). CONCLUSIONS: Cigarette smoking may be an independent, modifiable risk factor for non-insulin dependent diabetes mellitus. Moderate alcohol consumption among healthy people may be associated with increased insulin sensitivity and a reduced risk of diabetes.
disease: **Breast and Colorectal Cancer**
BACKGROUND: Increasing evidence suggests that diabetes mellitus (DM) is associated with increased cancer incidence and mortality. Several mechanisms involved in diabetes, such as promotion of cell proliferation and decreased apoptosis, may foster carcinogenesis. This study investigated the association between DM and cancer incidence and cancer-specific mortality in patients with breast and colorectal carcinoma. [...] The overall HR for breast cancer incidence was 1.23 (95 per cent confidence interval 1.12 to 1.34) and that for colorectal cancer was 1·26 (1·14 to 1·40) in patients with DM compared with those without diabetes. The overall HR was 1.38 (1.20 to 1.58) for breast cancer- and 1.30 (1.15 to 1.47) for colorectal cancer-specific mortality in patients with DM compared with those without diabetes. CONCLUSION: This meta-analysis indicated that DM is a risk factor for breast and colorectal cancer, and for cancer-specific mortality.

Table 2: Example of two paper abstracts manually annotated for risk factors. The highlighted text spans (comprising the factors) where marked by the co-author with medical background. Note that in some cases the precise name of the risk factor (e.g., "cigarette smoking") for a disease (e.g., "diabetes in men") is annotated in its broader context, to ensure the model is trained to extract risk factors tied to the disease, and not other, unrelated, artifacts.

class	P	R	F1
POS (with risk factor)	0.89	0.94	0.92
NEG (w/o risk factor)	0.94	0.89	0.92

Table 3: Classification results reported on the test set (20%) of the manually annotated 182 abstracts.

and operational effectiveness. In practice, when the model evaluates potential answers, it only considers text segments whose length does not exceed this predefined limit. Specifically, the text extracted between the predicted start and end indices is compared against the maximum length, and any text exceeding this threshold is disregarded.

Identification of Risk Factors at Scale Utilizing the fine-tuned QA model, we then processed the collected abstracts to identify and validate risk factors for a wide range of diseases, culminating in a dataset that catalogs these findings in much detail. As a concrete example, the entry for the "B-cell acute lymphoblastic leukemia" includes 16 (not necessarily unique) automatically extracted risk factors. Along with the extracted span, the BioBERT QA model provides its probability (confidence, in the 0-1 range) for the identified answer. For a given disease, we only considered answers exceeding the confidence of `0.6*max_answer_probability`, where the `max_answer_probability` is the maximum probability assigned to an answer for the disease. The final dataset encompasses the total of 162,409 identified risk factors spanning 744 diseases, extracted from 54,820 PubMed abstracts.

Due to the inherently strict nature of the `exact-match` metric, we could observe multiple cases where the extracted answer was largely correct, but didn't represent a precise overlap with the "gold" answer due to a single missing or redundant word. In particular, while some cases surface useful information about a disease risk factors, they are marked as inaccurate by the automatic metric. We complement the evaluation pipeline by sampling a large amount of (automatically identified) risk factors for diseases, and performing fine-grained human assessment of the results' quality.

5 Human Evaluation

We next manually evaluated a random sample of 1,485 extracted risk factors spanning 29 various diseases (constituting roughly 1% of the full set of extracted factors), based on their validity and relevance to the disease in question.

5.1 Evaluation Scheme

We designed a specifically-tailored, four-tiered annotation scheme for the sake of reliable and accurate evaluation, as detailed below. Each risk factor was scored with one of three annotation marks, following the below annotation scheme:

(1) Valid risk factor for the specified disease: Correctly identified risk factor extracted for the disease of interest, i.e., the disease in the question introduced to the QA system.

(2) Valid risk factor for a different disease: Correctly identified risk factor for a different disease, i.e., not the disease in the question introduced to the QA system, indicating capabilities yet highlighting challenges in specificity.

(3) Invalid risk factor: Phrases and terms that are not considered medical risk factors.

Additional distinction was done within the first group (valid risk factor), annotating risk factors with strong statistical correlation, as evident from the abstract by inspecting statistical measurements as odd ratio (OR), and confidence intervals (CIs) – metrics often used in medical literature for testing the significance of findings, such as the presence of a factor in one population but not the other. 41 out of the total of 1,485 were marked as *highly significant* risk factors; we release these annotations as well to facilitate further research in the community.

5.2 Evaluation Results

Table 4 presents error analysis of correctly- and incorrectly-identified risk factor examples (the first two rows), as well as an example for artifact that does not constitute a risk factor (the last row).

We attribute most factors erroneously annotated with type 3 annotation — not a risk factor — to cases where the QA model was required to extract a risk factor from an abstracts that does not contain one. Since the model was trained (and fine-tuned) to *always* identify an answer span for a given context and question, it is expected to yield (admittedly) weak performance on a context lacking the factors at the first place. Notably, a relatively small amount of all manually evaluated examples (around 8.5%) fall into this category.

Table 5 further summarizes the evaluation results by disease family. The prevalence of type 1 and 2 annotations illustrates the model's effectiveness in identifying risk factors, yet also underscores the challenges in achieving precise disease-specific accuracy. The presence of type 3 annotations, although significantly lower, highlights the ongoing need for the classification model refinement to enhance both specificity and accuracy.

Error Analysis Additional observation can be made about error distribution between type 1 and 2 annotations within and across disease families. Evidently, while some disease families show a balanced ratio between type 1 and 2 annotations (e.g., Infection, Leukemias), others resulted in more mis-identified factors – type 2 annotation (e.g., Metabolic disorders). We hypothesize that abstracts concerning diseases with a significant, sometimes absolute, genetic component are less likely to address other contributing factors. Consequently, research in this area predominantly focuses on stratifying potential risks for other diseases in individuals already affected by the genetic disorder.

6 Discussion and Limitations

Our study, while contributing valuable insights into the automation of risk factor identification from medical publications, is subject to several limitations that merit a thorough discussion.

One of the primary limitations is the challenge of accurately distinguishing risk factors specifically associated with the disease in question (type 1) from valid risk factors that are not directly related to the disease under investigation (type 2). While our models demonstrated a high capacity for identifying potential risk factors, the precision in contextualizing these factors to specific diseases varied. This aspect highlights a critical area for future research, emphasizing the need for enhanced specificity in the models to improve their utility in targeted medical research and practice.

Moreover, the study's reliance on free-text medical articles introduces variability in the data quality and representation. The unstructured nature of these texts and the diversity in how risk factors are described pose significant challenges for both the binary classification and question-answering models. Efforts to standardize data representation and improve model robustness against such variability are essential steps forward.

The datasets used in this study, while extensive, are not exhaustive. The landscape of medical research is continuously evolving, with new findings emerging regularly. The datasets, therefore, rep-

disease	abstract excerpt (identified risk factor highlighted)	marker
Chronic Myeloid Leukemia	[...] RESULTS: Previous diagnoses of dyspepsia, gastritis or peptic ulcers, as well as previous proton pump inhibitor (PPI) medication, were all associated with a significantly increased risk of CML (RRs, 1.5-2.0; P = 0.0005-0.05). Meanwhile, neither inflammatory bowel disease nor intake of NSAIDs were associated with CML, indicating that it is not gastrointestinal ulcer or inflammation per se that influences risk. [...]	1
Cystic Fibrosis	BACKGROUND: Cystic fibrosis, like other chronic diseases, is a risk factor for the development of elevated symptoms of depression and anxiety. [...] Patient anxiety (OR 2.33) and depression (OR 4.09) were significantly associated with forced expiratory volume in one second (FEV1) <40% and forced vital capacity (FVC) <80% (OR 1.60 and 1.61, respectively). CONCLUSIONS: Cystic fibrosis increases the risk of developing anxiety and depression in female patients and in mothers.	2
Renal Cell Carcinoma	RESULTS: A total of 888 incident RCCs and 356 RCC deaths were identified. In models including adjustment for body mass index and energy intake, there was no higher risk of incident RCC associated with consumption of juices (HR per 100 g/day increment = 1.03; 95% CI, 0.97-1.09), total soft drinks (HR = 1.01; 95% CI, 0.98-1.05), [...] CONCLUSIONS: Consumption of juices or soft drinks was not associated with RCC incidence or mortality after adjusting for obesity.	3

Table 4: Examples for automatic identification of risk factors in medical abstracts, marked by the annotator: 1 (valid risk factor for the specified disease) – stomach diseases are risk factors for CML; 2 (valid risk factor for a different disease) – CF, the disease of interest, was found to be a risk factor for depression and anxiety; and 3 (not a risk factor) – juices were **not** identified as a risk factor for RCC.

family (sub-family)	(1) valid risk factor for the specified disease	(2) valid risk factor for a different disease	(3) not a risk factor	total in family
Carcinomas	317	285	60	662
Infection	45	51	6	102
Leukemias	208	192	46	446
Lymphomas	27	12	4	43
Metabolic disorders (GD)	4	60	8	72
Mucus malefunction (GD)	11	34	2	47
Cardiomyopathy	5	23	0	28
Sarcomas	15	5	1	21
other hematological disorders	30	32	2	64
total	662	694	129	1485

Table 5: Distribution of manual evaluation annotations by disease family. "GD" denotes "genetic disorder". Note the much high number of risk factors identified for common (and potentially fatal) diseases, due to the vast body of empirical literature. The numbers refer to the total number of (not necessarily unique) risk factors identified for a disease family. We hypothesize that abstracts concerning diseases with a significant, sometimes absolute, genetic component are less likely to address other contributing factors; between the dashed lines in the table.

resent a snapshot in time, and ongoing efforts to update and expand these resources are necessary to maintain their relevance and utility.

Finally, the study's scope was constrained by the computational resources available. Future work could explore more complex models or ensemble approaches that might offer improved accuracy but require more substantial computational power.

Despite these limitations, this study represents a significant step toward automating the identification of disease risk factors from medical literature. Acknowledging and addressing these limitations in future research will be crucial for advancing the field and enhancing the practical applicability of these technologies in healthcare.

7 Conclusions and Future Work

This study presented an approach to identifying and extracting disease risk factors from free-text medical articles using advanced natural language processing techniques, specifically leveraging the capabilities of the pre-trained BioBERT-based architecture. Our methodology involved a multi-step process, including the retrieval of relevant articles, binary classification to filter articles discussing risk factors, and a question-answering model to extract specific risk factor information.

We have demonstrated the potential of language technologies to significantly enhance the efficiency and effectiveness of risk factor identification in medical literature. Our contributions to this field are twofold: the presentation of an automated pipeline for risk factor extraction and the creation of valuable datasets for future research. While our study marks an advancement in the automated extraction of risk factors from medical literature, there remain several avenues for future research and development. Our future directions include introducing improvements to QA model's accuracy and specificity, integration of additional data sources, and evaluation of more advanced LLMs for the task of risk factors identification.

Furthermore, inspired by recent findings that automatic annotations generated by models like GPT-4 can achieve results comparable to human annotations, we plan to investigate the use of GPT-4 for the task of risk factors annotation, and compare its performance with human experts.

8 Ethical Considerations

We make use of publicly available data in the domain of healthcare, that have been broadly used in numerous studies. Manual annotations were conducted by one of the authors of the paper, with medical background. Due to the required expertise and the inherent difficulty of the task, the mean hourly rate for the annotator was much higher than the established minimum wage.

Acknowledgements

We are grateful to Noga Shraibman for much help creating datasets during the early stages of this work. We are also thankful to our three anonymous reviewers for their constructive feedback.

References

Abdelaziz A Abdelhamid, Marwa M Eid, Mostafa Abotaleb, SK Towfek, et al. 2023. Identification of cardiovascular disease risk factors among diabetes patients using ontological data mining techniques. *Journal of Artificial Intelligence and Metaheuristics*, 4(2):45–53.

Svetla Boytcheva, Ivelina Nikolova, Galia Angelova, and Zhivko Angelov. 2017. Identification of risk factors in clinical texts through association rules. In *BiomedicalNLP@ RANLP*, pages 64–72.

Hao Chen and Burt M Sharp. 2004. Content-rich biological network constructed by mining pubmed abstracts. *BMC bioinformatics*, 5:1–13.

Qingcai Chen, Haodi Li, Buzhou Tang, Xiaolong Wang, Xin Liu, Zengjian Liu, Shu Liu, Weida Wang, Qiwen Deng, Suisong Zhu, et al. 2015. An automatic system to identify heart disease risk factors in clinical texts over time. *Journal of biomedical informatics*, 58:S158–S163.

Thanat Chokwijitkul, Anthony Nguyen, Hamed Hassanzadeh, and Siegfried Perez. 2018. Identifying risk factors for heart disease in electronic medical records: A deep learning approach. In *Proceedings of the BioNLP 2018 workshop*, pages 18–27.

Mary Rajathei David and Selvaraj Samuel. 2012. Clustering of pubmed abstracts using nearer terms of the domain. *Bioinformation*, 8(1):20.

Gibong Hong, Yuheun Kim, YeonJung Choi, and Min Song. 2021. Bioprep: deep learning-based predicate classification with semmeddb. *Journal of biomedical informatics*, 122:103888.

Ioannis Kavakiotis, Olga Tsave, Athanasios Salifoglou, Nicos Maglaveras, Ioannis Vlahavas, and Ioanna Chouvarda. 2017. Machine learning and data mining methods in diabetes research. *Computational and structural biotechnology journal*, 15:104–116.

Halil Kilicoglu, Graciela Rosemblat, Marcelo Fiszman, and Dongwook Shin. 2020. Broad-coverage biomedical relation extraction with semrep. *BMC bioinformatics*, 21:1–28.

Halil Kilicoglu, Dongwook Shin, Marcelo Fiszman, Graciela Rosemblat, and Thomas C Rindflesch. 2012. Semmeddb: a pubmed-scale repository of biomedical semantic predications. *Bioinformatics*, 28(23):3158–3160.

Florian Kronenberg, Samia Mora, Erik SG Stroes, Brian A Ference, Benoit J Arsenault, Lars Berglund, Marc R Dweck, Marlys Koschinsky, Gilles Lambert, François Mach, et al. 2022. Lipoprotein (a) in atherosclerotic cardiovascular disease and aortic stenosis: a european atherosclerosis society consensus statement. *European heart journal*, 43(39):3925–3946.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Ling Luo, Po-Ting Lai, Chih-Hsuan Wei, Cecilia N Arighi, and Zhiyong Lu. 2022. Biored: a rich biomedical relation extraction dataset. *Briefings in Bioinformatics*, 23(5):bbac282.

Awais Mehmood, Munwar Iqbal, Zahid Mehmood, Aun Irtaza, Marriam Nawaz, Tahira Nazir, and Momina Masood. 2021. Prediction of heart disease using deep convolutional neural networks. *Arabian Journal for Science and Engineering*, 46(4):3409–3422.

Yusuke Miyazawa, Narimasa Katsuta, Tamaki Nara, Shuko Nojiri, Toshio Naito, Makoto Hiki, Masako Ichikawa, Yoshihide Takeshita, Tadafumi Kato, Manabu Okumura, et al. 2024. Identification of risk factors for the onset of delirium associated with covid-19 by mining nursing records. *Plos one*, 19(1):e0296760.

Aakanksha Naik, Sravanthi Parasa, Sergey Feldman, Lucy Lu Wang, and Tom Hope. 2021. Literature-augmented clinical outcome prediction. *arXiv preprint arXiv:2111.08374*.

Kevin Roitero, Beatrice Portelli, Mihai Horia Popescu, and Vincenzo Della Mea. 2021. Dilbert: Cheap embeddings for disease related medical nlp. *IEEE Access*, 9:159714–159723.

Susan Sabra, Khalid Mahmood, and Mazen Alobaidi. 2017. A semantic extraction and sentimental assessment of risk factors (sesarf): an nlp approach for precision medicine: a medical decision support tool for early diagnosis from clinical notes. In *2017 IEEE 41st Annual Computer Software and Applications Conference*, volume 2, pages 131–136. IEEE.

Seyedmostafa Sheikhalishahi, Riccardo Miotto, Joel T Dudley, Alberto Lavelli, Fabio Rinaldi, Venet Osmani, et al. 2019. Natural language processing of clinical notes on chronic diseases: systematic review. *JMIR medical informatics*, 7(2):e12239.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, et al. 2023. Towards expert-level medical question answering with large language models. *arXiv preprint arXiv:2305.09617*.

Amber Stubbs, Christopher Kotfila, Hua Xu, and Özlem Uzuner. 2015. Identifying risk factors for heart disease over time: Overview of 2014 i2b2/uthealth shared task track 2. *Journal of biomedical informatics*, 58:S67–S77.

Christiaan H Vinkers, Joeri K Tijdink, and Willem M Otte. 2015. Use of positive and negative words in scientific pubmed abstracts between 1974 and 2014: retrospective analysis. *Bmj*, 351.

Xi Yang, Aokun Chen, Nima PourNejatian, Hoo Chang Shin, Kaleb E Smith, Christopher Parisien, Colin Compas, Cheryl Martin, Anthony B Costa, Mona G Flores, et al. 2022. A large language model for electronic health records. *NPJ digital medicine*, 5(1):194.

A Appendices

A.1 Overview of the Risk Factor Annotation System Architecture

The risk factor annotation system comprises three main components designed to streamline the process of annotating risk factors in medical articles. This system was instrumental in creating the datasets used in our research.

GraphQL Server The backbone of the system is a GraphQL server, which serves as the central communication hub. Hosted on Kubernetes (k8s) for scalability and reliability, the server facilitates data exchange between the user interface and the database. It handles requests for data retrieval and submission, ensuring that the web application and the code can access and store data efficiently.

Web UI The front end of the system is a React-based web application, also deployed on Kubernetes for high availability. This intuitive user interface allows medical students and researchers to interact with the system, including retrieving medical articles, annotating risk factors within texts, and submitting these annotations back to the server. The design prioritizes ease of use to facilitate accurate and efficient annotation work.

Python Algorithm Complementing the user interface is a Python-based algorithm that interacts with the GraphQL server. This component is responsible for processing medical articles, including sending requests to the server to fetch articles for annotation and submitting the results of automated risk factor identification processes. It plays a critical role in pre-processing and post-processing steps in the dataset creation pipeline.

Database At the core of the system lies a MongoDB database hosted on Azure Cosmos DB. This NoSQL database was chosen for its scalability, flexibility, and robust support for storing unstructured data, such as medical article texts and annotations. It stores all data related to diseases, articles, and user annotations, providing a persistent and reliable data storage solution for the system.

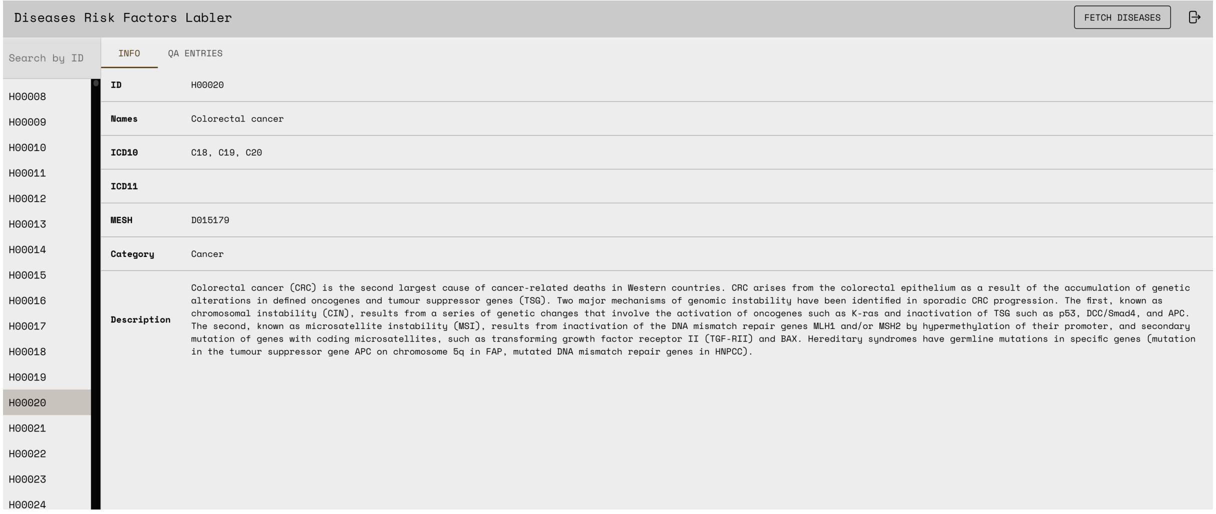
Figures 2-3 illustrate two screenshots of the application developed for manual annotation of risk factors. The system code will also be made available per acceptance.

B Diseases with Annotated Risk Factors in the QA dataset (the training set)

Section 3.3 details the procedure of manual annotation of risk factors following the step of abstract retrieval. The annotated data comprises 1,712 QA items from 668 abstracts covering 15 diseases from multiple disease families, as detailed in Table 6.

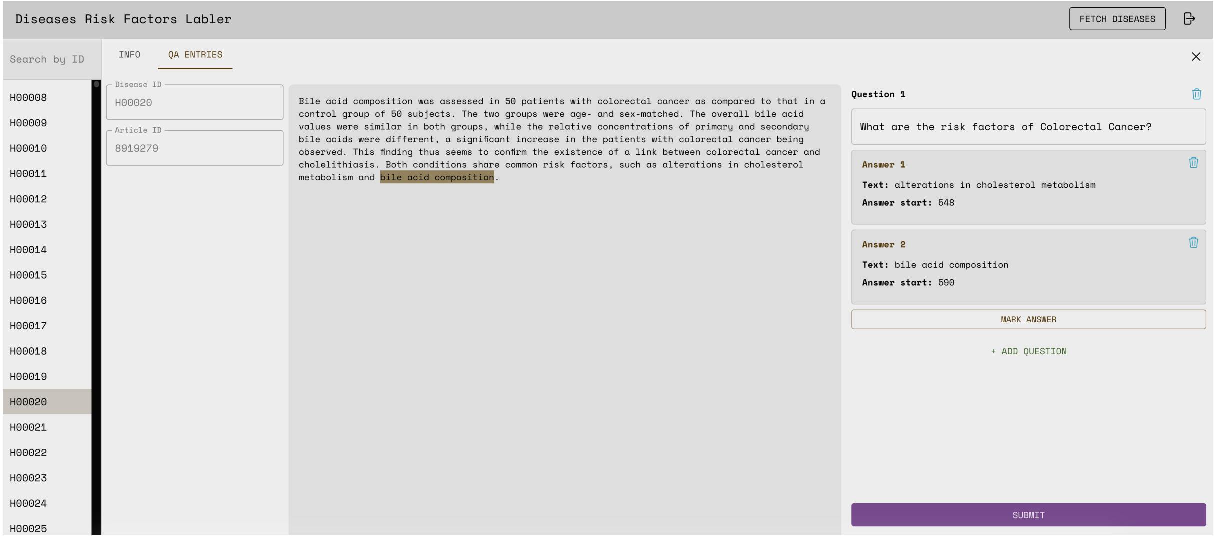
C Diseases with Evaluated Risk Factors

Table 7 reports the distribution of manually evaluated risk factors by disease family.



ID	H00020
Names	Colorectal cancer
ICD10	C18, C19, C20
ICD11	
MESH	D015179
Category	Cancer
Description	Colorectal cancer (CRC) is the second largest cause of cancer-related deaths in Western countries. CRC arises from the colorectal epithelium as a result of the accumulation of genetic alterations in defined oncogenes and tumour suppressor genes (TSG). Two major mechanisms of genomic instability have been identified in sporadic CRC progression. The first, known as chromosomal instability (CIN), results from a series of genetic changes that involve the activation of oncogenes such as K-ras and inactivation of TSG such as p53, DCC/Smad4, and APC. The second, known as microsatellite instability (MSI), results from inactivation of the DNA mismatch repair genes MLH1 and/or MSH2 by hypermethylation of their promoter, and secondary mutation of genes with coding microsatellites, such as transforming growth factor receptor II (TGF-RII) and BAX. Hereditary syndromes have germline mutations in specific genes (mutation in the tumour suppressor gene APC on chromosome 5q in FAP, mutated DNA mismatch repair genes in HNPCC).

Figure 2: Disease Risk Factor Annotation System: disease details as retrieved from KEGG and parsed.



Disease ID: H00020

Article ID: 8919279

Bile acid composition was assessed in 50 patients with colorectal cancer as compared to that in a control group of 50 subjects. The two groups were age- and sex-matched. The overall bile acid values were similar in both groups, while the relative concentrations of primary and secondary bile acids were different, a significant increase in the patients with colorectal cancer being observed. This finding thus seems to confirm the existence of a link between colorectal cancer and cholelithiasis. Both conditions share common risk factors, such as alterations in cholesterol metabolism and bile acid composition.

Question 1

What are the risk factors of Colorectal Cancer?

Answer 1

Text: alterations in cholesterol metabolism

Answer start: 548

Answer 2

Text: bile acid composition

Answer start: 590

Figure 3: Disease Risk Factor Annotation System: manual annotation of spans containing risk factors; multiple risk factors for the same disease can be identified in the same abstract.

family	disease
Autoimmune disease	Celiac disease
Autoimmune disease	Rheumatoid arthritis
Autoimmune disease	Type 1 diabetes mellitus
Carcinomas	Bladder cancer
Carcinomas (to the most part)	Breast cancer
Carcinomas (to the most part)	Colorectal cancer
Chronic lung disease	Chronic obstructive pulmonary disease
Chronic lung disease	Asthma
Circulatory disorder	High blood pressure
Heart disease	Myocardial infarction
Melanoma/Skin cancer	Melanoma
Metabolic disease	Metabolic syndrome
Metabolic disease	Type 2 diabetes mellitus
Neurodegenerative disorder	Alzheimer disease
Neurologic disorder	Migraine

Table 6: Disease distribution by disease family in the manually annotated set of 1,712 risk factors used for BioBERT QA fine-tuning.

family	disease
other hematological disorders	Multiple myeloma
Carcinomas	Choriocarcinoma
Carcinomas	Esophageal cancer
Carcinomas	Gastric cancer
Carcinomas	Malignant pleural mesothelioma
Carcinomas	Non-small cell lung cancer
Carcinomas	Penile cancer
Carcinomas	Renal cell carcinoma
Carcinomas	Small cell lung cancer
Carcinomas	Vulvar cancer
Infection	Cholera
infection	Gonococcal infection
infection	Pertussis
Leukemias	Acute myeloid leukemia
Leukemias	Adult T-cell leukemia
Leukemias	B-cell acute lymphoblastic leukemia
Leukemias	Chronic lymphocytic leukemia
Leukemias	Chronic myeloid leukemia
Leukemias	Hairy cell leukemia
Leukemias	Polycythemia vera
Leukemias	T-cell acute lymphoblastic leukemia
Lymphomas	Burkitt lymphoma
Lymphomas	Lymphoplasmacytic lymphoma
Metabolic disorders (GD)	Congenital adrenal hyperplasia
Metabolic disorders (GD)	Gaucher disease
Metabolic disorders (GD)	Hemochromatosis
Mucus malefunction (GD)	Cystic fibrosis
Cardiomyopathy	Dilated cardiomyopathy
Sarcomas	Osteosarcoma

Table 7: Disease distribution by disease family in the manually evaluated set of 1,485 identified risk factors. "GD" denotes "genetic disorder".

Intervention extraction in preclinical animal studies of Alzheimer's Disease: Enhancing regex performance with language model-based filtering

Yiyuan Pu[1], **Kaitlyn Hair**[3], **Daniel Beck**[2], **Mike Conway**[1],
Malcolm Macleod[3], **Karin Verspoor**[1,2]

[1] School of Computing and Information Systems, The University of Melbourne, Melbourne, Australia
[2] School of Computing Technologies, RMIT University, Melbourne, Australia
[3] Centre for Clinical Brain Sciences, The University of Edinburgh, Edinburgh, U.K

Abstract

We explore different information extraction tools for annotation of interventions to support automated systematic reviews of preclinical AD animal studies. We compare two PICO (Population, Intervention, Comparison, and Outcome) extraction tools and two prompting-based learning strategies based on Large Language Models (LLMs). Motivated by the high recall of a dictionary-based approach, we define a two-stage method, removing false positives obtained from regexes with a pre-trained LM. With ChatGPT-based filtering using three-shot prompting, our approach reduces almost two-thirds of False Positives compared to the dictionary approach alone, while outperforming knowledge-free instructional prompting.

1 Introduction and Related Work

Biomedical information extraction is the task of automatically extracting entities, relations, and events from biomedical literature (Hobbs, 2002; Liu et al., 2016). This information is in turn relevant to writing of systematic reviews, which support evidence-based decision making by identifying, integrating, and assimilating relevant articles on a given clinical question (Methley et al., 2014). A standard framework used for defining review questions is PICO, standing for Population (or Patient), Intervention (or Exposure), Comparison, and Outcome (Cooke et al., 2012).

We examine information extraction in Alzheimer's Disease (AD), which has affected more than 55 million people around the world[1]. We focus on detecting the PICO dimension of *Intervention* in the AD literature, where interventions are typically drugs. This task is sometimes referred to as *intervention extraction*. It suffers from having low precision compared to extraction of other PICO elements (Hair et al., 2023a). More precise extraction of interventions will support more effective systematic reviewing, and can help to prioritize drugs for clinical trials in literature-based discovery (Pu et al., 2023).

Standard methods for intervention extraction include dictionary-based approaches (Hair et al., 2023b) and machine learning models (Wang et al., 2021; Wei et al., 2024). The recent advent of generative Large Language Models (LLMs) and the prompting-based paradigm for information extraction (Liu et al., 2023) raise questions of how to better leverage them in this task and whether they outperform previous methods. This is particularly interesting in domain-specific scenarios such as AD, where limited data is available for training models (Wang et al., 2023). We address these questions, with two main contributions: 1) we show that while generative LLMs improve intervention extraction precision, they suffer from low recall compared to dictionary-based methods, and 2) we propose a two-stage architecture combining both dictionaries and LLMs that better balances precision and recall and reaches a new state-of-the-art on the AD dataset.

2 Methods

2.1 Data

We used a manually-curated dataset containing preclinical animal studies in the context of AD (Hair et al., 2023b). This dataset consists of documents comprising title, abstract, and keyword fields for 100 studies. The dataset was created in two steps: 1) a set of regular expression (regex) patterns corresponding to a dictionary of interventions was applied to annotate intervention entities, and 2) a human annotator labeled each tagged entity as "intervention" or "not an intervention". Figure 1 shows an example of an annotated document, the extracted entities, and the human judgment label for each. The AD dataset may not be perfect since

[1] https://www.alzint.org/about/dementia-facts-figures/dementia-statistics/

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 486–492
August 16, 2024. ©2024 Association for Computational Linguistics

	Training set	Test set
Document count	5	95
#Intervention	6	67
#Not an intervention	14	288

Table 1: AD dataset statistics for training and testing.

human annotations were a subset of regex annotations. Intervention entities not being captured by the regex dictionary were out of the scope for the human annotator. However, we used human annotations as a gold standard for this study.

Article id: PMID 31190768

Document: ["... Icariin (ICA) as one of the active ingredients of Chinese herbal medicine has the immunomodulating function. This study aimed to investigate the immunotherapeutic potential of ICA on AD... Then the ethological and biochemical experiments such as Morris water maze assay Aβ ELISA blood T cell flow cytometry and plasma and brain cytokines array were conducted to evaluate the effects of ICA administration. ..."]

Matched spans: [[156,163], [527,532]]

Text spans: ["Icariin", "water"]

Human labels: ["intervention", "not an intervention"]

Figure 1: An example of an annotated document in the dataset from Hair et al. (2023b). The "document" field contains the title, abstract, and keywords for each paper. Only part of the abstract is shown for brevity.

We randomly split the dataset into a training set (5 documents) and a test set (95 documents) (Table 1). The training documents were used for few-shot learning in prompt-based methods. All results are reported on the test set.

2.2 Baselines

We adapted three biomedical entity extraction tools for intervention extraction to use as baselines, detailed in Table 2. The regex-based method of (Hair et al., 2023b) utilizes a customized dictionary based on regular expressions for preclinical AD animal studies. Each publication was tagged with animal models, outcomes, interventions, species, and sexes; here we considered only entities tagged as interventions. The intervention dictionary had a list of 12,447 compounds compiled from DrugBank[2] and Alzforum[3]. Synonyms, alternate spellings, and punctuation differences were captured in regexes (Hair et al., 2023b). This method was used to create the dataset employed in our experiments, resulting in maximum recall by design.

Wang et al. (2021) constructed a PICO extraction workflow based on Bidirectional Encoder Representations (BERT; Devlin et al. (2019)) for general preclinical animal studies (not specific to AD). This method had two entity categories that relate to interventions: Intervention and Comparator. Intervention was defined as interventions that reflect clinical practice, while Comparator was defined as a control group, such as no treatment, vehicle/-placebo, sham treatment, or another intervention. We treated entities tagged as either Intervention or Comparator entity types as interventions.

Finally, we also used the latest version of PubTator 3.0 (Wei et al., 2024) as an additional baseline, due to its widespread usage in biomedical information extraction. This tool extracted proteins, genetic variants, diseases, and chemicals with a recently developed named entity recognition (NER) model called AIONER (Luo et al., 2023). We treated Pubtator-identified Chemical entities as Interventions. For this entity type, training was based on the NLM-Chem corpus (Islamaj et al., 2021), with ~5000 unique drug/chemical name annotations in 150 PubMed full-text chemical literature. The PubTator API[4] was used to conduct raw processing of input texts for entity extraction.

2.3 Prompt-based methods

Since we had limited labeled data, we prioritized prompt-based models over training machine learning or deep learning models. We followed the framework of (Liu et al., 2023) to design our prompt-based models. We considered four aspects of prompt-based learning for intervention extraction: pre-trained language models (PLMs), prompting templates, answer space, and prompting parameters.

Pre-trained LMs We selected ChatGPT[5] and GPT-4 (OpenAI, 2023) as the PLMs.[6]

Prompting templates We adapted prompting templates previously used for zero-shot gene extraction in biomedical literature (Törnkvist, 2024) to our task for both zero-shot and few-shot learning.

[2] https://go.drugbank.com/drugs

[3] https://www.alzforum.org/therapeutics

[4] https://www.ncbi.nlm.nih.gov/research/pubtator/api.html

[5] https://platform.openai.com/docs/models/gpt-3-5-turbo

[6] In an effort to employ open-source LMs, we also considered OLMo (Groeneveld et al., 2024). However, we were not able to obtain meaningful answers from this LM.

Entity extraction tool	Scope of entity types	Entity types used
Regex-based method (Hair et al., 2023b)	Animal model, Outcome measure, Intervention, Species, Sex	Intervention
BERT-based method (Wang et al., 2021)	Intervention, Comparator, Outcome, Species, Strain, Induction	Intervention, Comparator
PubTator 3.0 (Wei et al., 2024)	Gene/Protein, Variant, Disease, Chemical, Species, and Cell Line	Chemical
ChatGPT (OpenAI, 2023)	-	Intervention
GPT-4 (OpenAI, 2023)	-	Intervention

Table 2: Scope and use of entity types for this study from entity extraction tools

Our templates[7] are described in Appendix B.

Answer space We used text spans from the documents that were recognized as interventions.

Prompting parameters For both models, we set temperature to be 0.7, max_tokens as 50, and top_p as 1. The "temperature" parameter controls randomness, which ranges between 0 to 2.

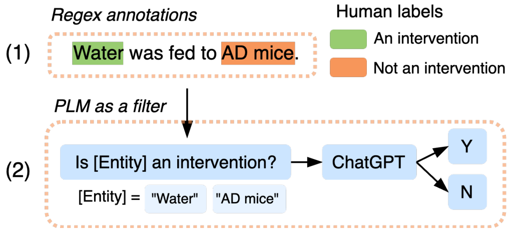


Figure 2: An example of the two-stage filtering method that we proposed. (1) A regex-based method annotated potential interventions in each document. Each potential intervention had a corresponding human label, indicating whether it was an intervention in context. (2) Each potential intervention and its context were inputs for a PLM. Figure is simplified due to space limitations.

2.4 Two-stage filtering method

The approach we proposed (Figure 2) was motivated by the maximum recall provided by the regex patterns used to create the dataset. Instead of having a PLM doing the full work of extracting interventions, we proposed using it to *filter the false positives obtained from regexes*. Precision errors arising from regex-based methods were mainly due to a lack of context: any entity that matched a regex would be recognized as an intervention. We hypothesized that a PLM can contextualize entity context and filter them out if appropriate, without undermining recall.

We experimented with both zero-shot and few-shot approaches (with examples sampled from the training set), using ChatGPT as the PLM. We tailored our prompts to frame the task as filtering (described in Appendix C). All other parameters were the same as described in Section 2.3.

3 Results and Discussion

Table 3 summarises our results, reporting precision, recall, and F1 scores for all methods. As expected, the regex-based method resulted in perfect recall (since it was employed to develop the dataset in the first place) but low precision. Both BERT-based and PubTator 3.0 approaches did not perform well, likely due to domain differences. The prompt-based methods resulted in slightly better precision, at the price of a large decrease in recall.

Our zero-shot filtering approach outperformed the baselines in F1 score, with only a small decrease in recall. It filtered 30% of the false positives (FPs) of the regex-based method. A three-shot variant, adding three true positive (TP) examples to the prompt, gave even better precision and F1 score, filtering almost two-thirds of all FPs.

3.1 Further analysis on few-shot prompting

We performed additional experiments analyzing the influence of adding positive examples to the prompt in our two-stage method. The regex-only baseline resulted in 67 TPs and 288 FPs: an ideal filtering layer should remove all FPs while keeping the original TPs.

Table 4 shows detailed results using two metrics: the total reduction in FPs and the total reduction in TPs (the latter framed as "TP price", since this should ideally be zero). In general, the higher the FP reduction, the higher the TP price. However, we did not see any particular trends when increasing

[7] The prompting templates (Appendix B) used for baselines were different from the templates (Appendix C) in the two-stage filtering method. For baselines, the prompting templates were adapted from Törnkvist (2024). For the two-stage filtering method, the prompting templates were created on our own.

Intervention extraction method	**TP**	**FP**	**FN**	**Precision**	**Recall**	**F1 score**
Baselines						
Regex-based	**67**	288	**0**	0.19	**1.00**	0.32
BERT-based	22	232	45	0.09	0.33	0.14
PubTator 3.0	43	369	24	0.10	0.64	0.18
ChatGPT (0-shot)	28	97	39	0.22	0.42	0.29
ChatGPT (3-shot)	24	**61**	43	0.28	0.36	0.32
GPT-4 (0-shot)	27	108	40	0.20	0.40	0.27
GPT-4 (3-shot)	30	94	37	0.24	0.45	0.31
Our approach						
Regex+ChatGPT (0-shot)	64	203	3	0.24	0.96	0.38
Regex+ChatGPT (3-shot)	58	107	9	**0.35**	0.87	**0.50**

Table 3: Results for all intervention extraction. The first three columns detail true positives (TPs), false positives (FPs) and false negatives (FNs), while the last three columns report our evaluation metrics.

the number of examples, except for an outlier result for 1-shot prompting. Our 3-shot results provided the best balance but more work is required to understand if further increasing the number of examples can result in better performance.

#Examples	**TP**	**FP**	**TP price**	**FP reduction**
Baseline	67	288	0 (0%)	0 (0%)
0	64	203	3 (4%)	85 (30%)
1	45	90	22 (33%)	198 (69%)
2	63	181	4 (5%)	107 (37%)
3	58	107	9 (13%)	181 (62%)
4	63	177	4 (5%)	111 (38%)
5	61	186	6 (8%)	102 (35%)

Table 4: Detailed results varying the number of examples using our two-stage approach. All examples were "positive" labels (entities labeled as interventions by the human annotator), sampled from the training set.

We also tried using negative examples for few-shot learning. However, this did not improve the performance compared to using positive examples only. We report detailed results in Appendix A.

3.2 Motivating case study

We discuss case studies for our motivation for employing the two-stage filtering method. We analyzed the False Positives (FPs) of the regex-based method. As this method annotated every text span matching the intervention dictionary indiscriminately, the 288 FPs came from context recognition errors, i.e. where an intervention term did not describe a relevant intervention in the context of a document. For instance, the potential intervention entity "quercetin" in PMID:36840284 (Table 5) was part of the molecular modeling results of a compound rather than a drug whose effects were directly studied.

PMID	36840284
Evaluation	False Positive
Context	"Molecular modeling results revealed that the compound's ellagic acid, epicatechin, catechin, kaempferol, quercetin, and apigenin have the potential to act as a dual inhibitor of acetylcholinesterase (AChE) and COX-2 and can be responsible for the improvement of both cholinergic and inflammatory conditions."
PMID	30618732
Evaluation	True Positive
Context	"This study aimed to evaluate the neuroprotective effect of quercetin against the detrimental effects of LPS such as neuroinflammation-mediated neurodegeneration and synaptic/memory dysfunction in adult mice."

Table 5: An example of the same entity string labeled as both an intervention and not an intervention in distinct contexts.

One may argue that removing "quercetin" from the regex dictionary would reduce FPs. However, the entity "quercetin" was also used as an intervention in other contexts. As shown in Table 5, PMID:30618732 assessed the effects of "quercetin" as an intervention for treating adult mice with neurodegenerative diseases. Therefore, an ideal method must contextually differentiate usages of the putative entity mentions.

3.3 PLM response outliers

A generative PLM may produce model responses out of the target answer space, requiring further processing. In the Regex+ChatGPT (0-shot) scenario, the model responded with "therapeutic" (cf. "intervention"/"not an intervention") for a potential intervention entity string "therapeutic" and a given context of PMID:25061594. In the Regex+ChatGPT (3-shot) scenario, the model responded with a copy of the prompting template for a potential intervention “potassium" for PMID:30548427.

For these two outliers, we reverted to the output of a RegEx phase, i.e. with the label “intervention”.

4 Conclusion

In this work, we proposed a two-stage approach for intervention extraction that combined a regex-based method with a filtering step done by prompting a generative LLM. This approach outperformed strong baselines, including standalone use of LLMs. Effectively, we show that LLMs can augment regex/dictionary-based methods by removing context recognition errors.

Future work involves extending our approach to all PICO entities, beyond just interventions. This will help automate important tasks in the literature review for AD, such as collecting data for systematic reviews, and support creating more precise knowledge graphs for literature-based discovery. The same approach could also be adapted with specific resources and be applied to other datasets and domains, such as clinical trials (Nye et al., 2018). Finally, different strategies to employ LLMs in the filtering step could be investigated, such as fine-tuning.

References

Alison Cooke, Debbie Smith, and Andrew Booth. 2012. Beyond PICO: The SPIDER Tool for Qualitative Evidence Synthesis. *Qualitative Health Research*, 22(10):1435–1443.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *North American Chapter of the Association for Computational Linguistics*.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Raghavi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, Will Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah A. Smith, and Hannaneh Hajishirzi. 2024. OLMo: Accelerating the Science of Language Models. *Preprint*, arXiv:2402.00838.

Kaitlyn Hair, Emma Wilson, Olena Maksym, Malcolm Robert Macleod, and Emily Sena. 2023a. A Systematic Online Living Evidence Summary of experimental Alzheimer’s disease research. *MetaArXiv*.

Kaitlyn Hair, Emma Wilson, Charis Wong, Anthony Tsang, Malcolm R Macleod, and Alexandra Bannach-Brown. 2023b. Systematic online living evidence summaries: emerging tools to accelerate evidence synthesis. *Clinical Science (London, England : 1979)*, 137:773 – 784.

Jerry R. Hobbs. 2002. Information extraction from biomedical text. *Journal of Biomedical Informatics*, 35(4):260–264. Sublanguage - Zellig Harris Memorial.

Rezarta Islamaj, Robert Leaman, Sun Kim, Dongseop Kwon, Chih-Hsuan Wei, Donald C. Comeau, Yifan Peng, David Cissel, Cathleen Coss, Carol Fisher, Rob Guzman, Preeti Gokal Kochar, Stella Koppel, Dorothy Trinh, Keiko Sekiya, Janice Ward, Deborah Whitman, Susan Schmidt, and Zhiyong Lu. 2021. NLM-Chem, a new resource for chemical entity recognition in PubMed full text literature. *Scientific Data*, 8.

Feifan Liu, Jinying Chen, Abhyuday N. Jagannatha, and Hong Yu. 2016. Learning for biomedical information extraction: Methodological review of recent advances. *ArXiv*, abs/1606.07993.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Comput. Surv.*, 55(9).

Ling Luo, Chih-Hsuan Wei, Po-Ting Lai, Robert Leaman, Qingyu Chen, and Zhiyong Lu. 2023. AIONER: all-in-one scheme-based biomedical named entity recognition using deep learning. *Bioinformatics*, 39(5):btad310.

Abigail M Methley, Stephen Campbell, Carolyn Chew-Graham, Rosalind McNally, and Sudeh Cheraghi-Sohi. 2014. PICO, PICOS and SPIDER: a comparison study of specificity and sensitivity in three search tools for qualitative systematic reviews. *BMC health services research*, 14(1):1–10.

Benjamin Nye, Junyi Jessy Li, Roma Patel, Yinfei Yang, Iain Marshall, Ani Nenkova, and Byron Wallace. 2018. A corpus with multi-level annotations of patients, interventions and outcomes to support language processing for medical literature. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 197–207, Melbourne, Australia. Association for Computational Linguistics.

OpenAI. 2023. GPT-4 Technical Report.

Yiyuan Pu, Daniel Beck, and Karin Verspoor. 2023. Graph embedding-based link prediction for literature-based discovery in Alzheimer’s Disease. *Journal of Biomedical Informatics*, 145:104464.

Betty Törnkvist. 2024. Named Entity Recognition for Detecting Trends in Biomedical Literature. Master's thesis, Umeå University, Department of Computing Science.

Benyou Wang, Qianqian Xie, Jiahuan Pei, Zhihong Chen, Prayag Tiwari, Zhao Li, and Jie Fu. 2023. Pre-trained Language Models in Biomedical Domain: A Systematic Survey. *ACM Comput. Surv.*, 56(3).

Qianying Wang, Jing Liao, Mirella Lapata, and Malcolm Macleod. 2021. PICO entity extraction for preclinical animal literature. *Systematic Reviews*, 11.

Chih-Hsuan Wei, Alexis Allot, Po-Ting Lai, Robert Leaman, Shubo Tian, Ling Luo, Qiao Jin, Zhizheng Wang, Qingyu Chen, and Zhiyong Lu. 2024. PubTator 3.0: an AI-powered literature resource for unlocking biomedical knowledge. *Nucleic Acids Research*.

A Prompting with negative shot for a two-stage filtering method

#Examples	#TP	#FP	TP price	FP reduction
Baseline	67	288	0 (0%)	0 (0%)
0	64	203	3 (4%)	85 (30%)
1	41	104	26 (38%)	184 (63%)
2	52	118	15 (22%)	170 (59%)
3	58	167	9 (13%)	121 (42%)
4	61	167	6 (8%)	121 (42%)
5	53	106	14 (20%)	182 (63%)

Table 6: Detailed results varying the number of examples using our two-stage approach. Selected examples were with human labels [“Positive", “Negative", “Positive", “Negative", “Positive"], sampled from the training set.

B A prompting template for intervention extraction baselines

B.1 Zero-shot learning

Task description: Please identify any mention of interventions in the text. Answer only the detected interventions and if more than one is found, separate them with ‘;’ not ‘and’. The answer should only contain the names of the interventions and nothing else. If no intervention is found, answer ‘None’.
Task content: Text: In this study we investigated the pharmacological influence of methylphenidate (MPH) on behavioral deficits of 5xFAD mice.

Table 7: An example of a prompting template for intervention extraction baselines (zero-shot). “Task description" is for the role of “system", while “Task content" is for the role of “user".

B.2 Few-shot learning

Task description: Please identify any mention of interventions in the text. Answer only the detected interventions and if more than one is found, separate them with ‘;’ not ‘and’. The answer should only contain the names of the interventions and nothing else. If no intervention is found, answer ‘None’.
Task content structure: <Examples for few-shot learning> Learn from the examples and complete the following task. <Text>
Task content: Here are examples for the task. The following is the first example. Text: Purpose: To study the effect of vitamin B2 (VB2) on the development of Alzheimer's disease (AD). Identified interventions in the text: vitamin; vitamin B2. Learn from the examples and complete the following task. Text: In this study we investigated the pharmacological influence of methylphenidate (MPH) on behavioral deficits of 5xFAD mice.

Table 8: An example of a prompting template for intervention extraction baselines (few-shot). “Task description" is for the role of “system".“Task content" is for the role of “user". “Task content structure" is a structure to create “Task content".

C A prompting template for a two-stage filtering method

C.1 Zero-shot learning

Task description: You will be provided with a text span and a block of text. Your task is to decide an entity type for the text span by considering the block of text as a context.
Task content: Text span [X]: glutathione. A block of text: Moreover the reduced activities or contents of glutathione reductase superoxide dismutase (SOD) and reduced GSH within the cortex and hippocampus caused by scopolamine were elevated by the treatment of KD-501. Please fill in the slot [Z]: [X] belongs to an entity type [Z]. Choose an entity type [Z] from the [‘intervention’, ‘not an intervention’]

Table 9: An example of a prompting template for a two-stage filtering method (zero-shot). “Task description" is for the role of “system", while “Task content" is for the role of “user".

C.2 Few-shot learning

Task description: You will be provided with a text span and a block of text. Your task is to decide an entity type for the text span by considering the block of text as a context.
Task content structure: <Examples for few-shot learning> Learn from the examples and complete the following task. <Text span> <A block of text> Please fill in the slot [Z]: [X] belongs to an entity type [Z]. Choose an entity type [Z] from the [‘intervention’, ‘not an intervention’]
Task content: Here are examples for the task. The following is the first example. Text span [X]: Quercetin. A block of text: Prosencephalon/metabolism/ultrastructure, Quercetin/*administration & dosage. [X] belongs to an entity type intervention. Learn from the examples and complete the following task. Text span [X]: glutathione. A block of text: Moreover the reduced activities or contents of glutathione reductase superoxide dismutase (SOD) and reduced GSH within the cortex and hippocampus caused by scopolamine were elevated by the treatment of KD-501. Please fill in the slot [Z]: [X] belongs to an entity type [Z]. Choose an entity type [Z] from the [‘intervention’, ‘not an intervention’]

Table 10: An example of a prompting template for a two-stage filtering method (few-shot). “Task description" is for the role of “system". “Task content" is for the role of “user". “Task content structure" is a structure to create “Task content".

Efficient Biomedical Entity Linking: Clinical Text Standardization with Low-Resource Techniques

Akshit Achara [*] **Sanand Sasidharan** [†] **Gagan N** [‡]

GE Healthcare

Abstract

Clinical text is rich in information, with mentions of treatment, medication and anatomy among many other clinical terms. Multiple terms can refer to the same core concepts which can be referred as a clinical entity. Ontologies like the Unified Medical Language System (UMLS) are developed and maintained to store millions of clinical entities including the definitions, relations and other corresponding information. These ontologies are used for standardization of clinical text by normalizing varying surface forms of a clinical term through Biomedical entity linking. With the introduction of transformer-based language models, there has been significant progress in Biomedical entity linking. In this work, we focus on learning through synonym pairs associated with the entities. As compared to the existing approaches, our approach significantly reduces the training data and resource consumption. Moreover, we propose a suite of context-based and context-less reranking techniques for performing the entity disambiguation. Overall, we achieve similar performance to the state-of-the-art zero-shot and distant supervised entity linking techniques on the Medmentions dataset, the largest annotated dataset on UMLS, without any domain-based training. Finally, we show that retrieval performance alone might not be sufficient as an evaluation metric and introduce an article level quantitative and qualitative analysis to reveal further insights on the performance of entity linking methods.

1 Introduction and Related Work

Medical text consists of a diverse vocabulary derived from various nomenclatures including varying surface forms corresponding to terms like diagnosis, treatment, medications, etc. This diversity poses a challenge for effective communication across medical institutions and organizations. One of the techniques to mitigate this inherent diversity present in multiple references to the same term is entity linking. Entity linking is used to map these references to standardized codes. These codes are curated and maintained by medical organizations for standardization of medical nomenclature.

Given a corpus, *entity linking* includes the mapping of a mention m which is a span of k words, to an entity ϵ, where the entity belongs to a knowledge base such as Wikipedia. In the biomedical domain, the textual phrases are linked with the corresponding concepts from a knowledge base constructed using the medical ontologies like UMLS (Bodenreider, 2004), SNOMED (El-Sappagh et al., 2018), etc. The UMLS ontology comprises of a broad range of clinical entities along with rich information for each entity like synonyms, definitions, etc. Traditional approaches for entity linking, such as Support Vector Machines (Cristianini and Shawe-Taylor, 2000) and Random Forests (Breiman, 2001), rely heavily on hand-crafted features, thereby restricting generalization to diverse data. Neural networks have emerged as a prominent technique for entity linking due to their ability to learn semantic representations from textual data.

Alias matching based techniques like (Aronson, 2001; Neumann et al., 2019; Liu et al., 2020) have been proposed where an input mention is mapped to an alias associated with an entity in the knowledge-base. However, these techniques require large amount of training data. Contextualized entity linking approaches (Zhang et al., 2021) utilize the semantic similarity between contextualized mentions. This approach requires a list of entities in advance and includes distant-supervision on articles containing examples of these entities. Generating medical codes using large language models can be error prone (Soroush et al., 2024). In (Yuan et al., 2022b), the authors use a seq2seq model to map a

[*] akshit.achara@gehealthcare.com
[†] sanand.sasidharan@gehealthcare.com
[‡] gagan.n@gehealthcare.com

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 493–505
August 16, 2024. ©2024 Association for Computational Linguistics

mention to its canonical entity name. This method is resource intensive and requires generation of synthetic examples for pretraining, utilizing entity definitions and synonyms. In (Kong et al., 2021), the authors propose a zero-shot entity linking approach by leveraging synonym and graph based tasks. However, the approaches require training samples from UMLS for both these tasks. Moreover, entity disambiguation has not been explored in the work.

Efficient student models like MiniLM (Wang et al., 2020) can be used to perform contrastive learning on synonyms of entities. This results in a significantly less embedding size (384) as compared to the approaches like SAPBERT (Liu et al., 2020) with an embedding size of 768. The predicted candidates in alias based techniques are ranked based on the cosine similarity score. However, there are ambiguous cases where multiple entities have similar scores for a common mention. Therefore, there is a requirement to disambiguate these candidates through reranking. Cross-Attention based reranking approaches utilize supervised training on the concatenated mention and candidate representations as inputs (Zhang et al., 2021). More recent approaches utilize homonym disambiguation (Garda and Leser, 2024) and have shown to improve the performance of autoregressive approaches like GenBioEL.

In comparison to the discussed techniques, we propose an efficient and low resource zero-shot biomedical entity linking approach along with a suite of disambiguation techniques. Furthermore, we introduce an article level similarity analysis to obtain further insights. This also allows us to conduct a qualitative analysis without manually going through all the articles manually.

Our contributions are as follows:

- **Data**: We show that the impact of training is negligible on a finetuned MiniLM model[1] as compared to the pretrained MiniLM model. Moreover, the pretrained MiniLM model when finetuned on all UMLS synonym pairs has worse performance than the all-MiniLM model.

- **Disambiguation** We show that reranking on entity-level semantic information provided in UMLS can be highly effective for entity disambiguation. We further propose a parametric reranking technique that is beneficial for alias-based entity linking solutions.

- **Evaluation** We propose a comprehensive evaluation of entity linking which utilizes the semantic representation of articles coupled with the strict matching and related matching of predicted and gold standard entities. This evaluation is used to highlight issues related to the annotation granularity, missing context and surface form bias (for abbreviations) without the need of going through all the articles.

2 Datasets

In this work, we explore entity linking on the Medmentions (Mohan and Li, 2019) dataset which consists of titles and abstracts from 4392 English biomedical articles. These articles comprise of textual spans annotated with mentions of UMLS 2017AA entities. The dataset provides two versions: a full version containing 34724 unique entities and an st21pv version with 25419 unique entities, the latter being recommended by the authors for information retrieval. Further details about the dataset versions are discussed in Table 9 in (Kartchner et al., 2023).

2.1 Preprocessing

We replace the abbreviations with their corresponding full forms using Ab3p (Sohn et al., 2008). The abbreviation expansion using Ab3p has shown to significantly improve the entity linking performance across different approaches (Kartchner et al., 2023). Prior to creating synonym pairs for training, we remove all the suppressed entities, deleted entities and deprecated entities. Some deprecated entities have also been merged with other entities having a synonymous relation. We map these deprecated entities to the corresponding active entities with a synonymous relation.

	st21pv	full
merged	181	280
deleted	49	60
non-synonymous	226	348

Table 1: The table shows the details of Medmentions entities annotated with UMLS 2017AA version that are deprecated in UMLS 2023AB version.

Some annotations in Medmentions (prepared with UMLS 2017AA) are deprecated in the UMLS

[1] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

2023AB (see details in table 1). Therefore, approaches utilizing UMLS 2023AB version may want to use an updated version of Medmentions. Furthermore, the prototype space (feature vector space) consisting of UMLS entities will have to be updated to the remove deprecated entities. This would help in avoiding deprecated entities to be predicted as candidates.

3 Methodology

In this work, we create a prototype vector space comprising of the encodings (feature vectors) associated with the canonical name of each entity in the UMLS ontology. To obtain meaningful encodings for constructing this prototype space, we train an encoder-based transformer (Vaswani et al., 2017) model on pairs of canonical names of entity synonyms. This is similar to the training approaches utilized in (Kong et al., 2021) and (Liu et al., 2020). The prototype space constructed using this trained model is used for performing semantic search, where the query encoding is obtained by passing the mention through the same model. This step is known as candidate generation. The candidate generation may lead to ambiguous results where multiple predicted entities have equal similarity scores. This is addressed through the reranking approaches discussed in section 3.3. Finally, we utilize both semantic similarity and retrieval performance for our quantitaive and qualitative evaluation. The comprehensive structure of our proposed approaches is depicted in the figure 1.

The following sub-sections discuss the individual components used in our work:

3.1 Training

We construct a training dataset by taking all the canonical names for each entity from UMLS and create pairs of canonical names corresponding to the same entity. Each pair is of the form (ϵ_i, ϵ_i^*), where ϵ_i^* represents the canonical name of a synonym of entity ϵ_i. The preprocessing steps are discussed in the section 2.1. We use this dataset to finetune a sentence-transformer (Reimers and Gurevych, 2019) model using Multiple Negatives Ranking loss (Henderson et al., 2017). We use MiniLM (Wang et al., 2020) which is a distilled version of BERT$_{BASE}$ model obtained using an effective knowledge distillation approach outperforming other lightweight models like TinyBERT and DistillBERT. We also utilize a finetuned all-MiniLM[2] model for training/finetuning on this dataset. The all-MiniLM model is obtained by training the MiniLM model on a 1B sentence pairs dataset using a contrastive learning objective. The corresponding MiniLM and all-MiniLM models trained/finetuned on k examples are hereafter referred as MiniEL$_k^*$ and MiniEL$_k$ respectively. For example, the all-MiniLM model finetuned on 10 pairs/examples is referred as MiniEL$_{10}$.

The Multiple Negatives Ranking loss function is defined as:

$$L(x, y, \theta) = \frac{1}{B}\sum_{j=1}^{B} \log P(y_j|x_j) \qquad (1)$$

Here, θ represents the network parameters, (x, y) represents a pair of phrases and B represents the batch size. The parameters details for training are provided in section in Appendix in the section A.1.

3.2 Candidate Generation

A prototype space is prepared for the UMLS 2017AA version comprising of the encodings of canonical names of each entity and its synonyms. These encodings are computed using the MiniEL* and MiniEL models. The prototype space is used for performing semantic search where the queries are formed using the labeled mentions from the Medmentions dataset. The top-k concepts are retrieved based on the cosine similarity of the query and entity encodings. These candidates are referred as generated candidates.

3.3 Disambiguation

The candidate generation solely relies on the cosine similarity score between the mention and prototype space candidate encodings. However, there may be cases where multiple candidates have similar scores or the scores alone may not be sufficient to rank the candidates. Therefore, there is a need to rerank the candidates. We propose the following reranking approaches that to perform the entity disambiguation:

3.3.1 Parametric Reranking

In this section, we propose a parametric approach to rerank the generated candidates. We consider three parameters based on the prototype space and our training framework for disambiguation namely, cosine similarity score (CSS), representative alias

[2] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

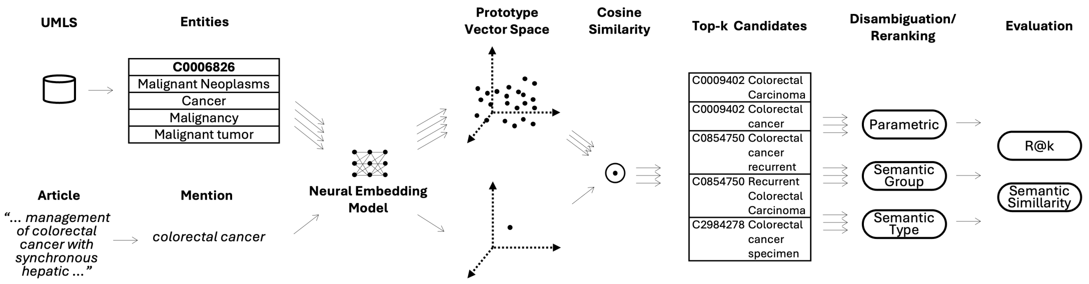

UMLS
Entities
C0006826
Malignant Neoplasms
Cancer
Malignancy
Malignant tumor
Article
"... management of colorectal cancer with synchronous hepatic ..."
Mention
colorectal cancer
Neural Embedding Model
Prototype Vector Space
Cosine Similarity
Top-k Candidates
C0009402 Colorectal Carcinoma
C0009402 Colorectal cancer
C0854750 Colorectal cancer recurrent
C0854750 Recurrent Colorectal Carcinoma
C2984278 Colorectal cancer specimen
Disambiguation/ Reranking
Parametric
Semantic Group
Semantic Type
Evaluation
R@k
Semantic Similarity

Figure 1: This figure illustrates the sequential flow of our proposed approaches. Starting from the left, we begin with leveraging a neural embedding model to create a prototype space on the UMLS entities. The cosine similarity metric is used to perform semantic search on the queries given the input mentions. The resultant top-k candidates are reranked using the listed methods for disambiguation and finally a comprehensive evaluation comprising of the retrieval performance and semantic similarity is performed.

score (RAS) and the candidate entity frequency score (CEFS) as our parameters. The parameters have the corresponding coefficients a, b and c respectively. These parameters are used to compute a new ranking score for each candidate. The equation below shows the updated score ($\delta^*(.,.)$) computation for reranking each of the generated candidates.

$$\delta^*(q,v) = a * \delta(q,v) + b * \frac{1}{n}\sum_{j=1}^{n} \delta(q, v_j) + c * n \quad (2)$$

Here, q is the query encoding, v is a generated candidate encoding and n is the number of aliases of v in the generated candidates.

The optimal selection of coefficients a, b and c corresponding to each of these parameters is performed through a grid search on a subset of manually defined bounds. Further details on the grid search and the impact of the these coefficients are discussed in appendix in the section A.2.

3.3.2 With UMLS Semantic Information

UMLS comprises of additional classification associated with individual entities, grouping them based on their semantic types and semantic groups. Each semantic type and semantic group has a canonical name. In this section, we calculate the cosine similarity between the mention's semantic type or semantic group canonical name encoding and the corresponding canonical names of the top-k candidates. This similarity score is added to the initial candidate generation score to rerank the top-k candidates.

1. **Assuming Availability of Gold Standard Information**: In this case, we assume that the gold standard semantic type and semantic group information is available for each mention. We rerank the candidates by utilizing the following methods:

   (a) **Semantic Type Based Disambiguation**: In this method, calculate the cosine similarity between canonical name encodings of semantic types of a mention and each of its top-k candidates. The updated score is computed as follows:

   $$\delta^*(q,v) = \delta(q,v) + \delta(TUI(q), TUI(v)) \quad (3)$$

   Here, $TUI(.)$ maps the input mention/entity to the encoding of corresponding semantic type canonical names.

   (b) **Semantic Group Based Disambiguation**: In this method, calculate the cosine similarity between canonical name encodings of semantic groups of a mention and each of its top-k candidates. The updated score is computed as follows:

   $$\delta^*(q,v) = \delta(q,v) + \delta(SG(q), SG(v)) \quad (4)$$

   Here, SG(.) maps the input mention/entity to the encoding of the corresponding semantic group canonical names.

2. **Semantic Type/Group Prediction**: In scenarios where the semantic type/group information of the mentions is not available, the methods proposed in (Le et al., 2022) and (Mao et al., 2023) can be used to predict the semantic type or group based on the input mentions. This can be followed by the computational methods discussed in the section 3.3.2.

4 Results and Discussion

We obtain the retrieval performance for the discussed approaches by considering the top-k closest candidates (that include aliases) from the prototype space. We observe that the retrieval performance (considering top-128 candidates) of all the miniEL and miniEL$_{1000}$ approaches is around 87% for the st21pv version and 88% for the full version of the Medmentions dataset.

4.1 Quantitative Analysis

In this section, we present the quantitive analysis associated with candidate generation (see section 4.1.1 and tables 2, 3) and reranking (see section 4.1.2, figure 2 and tables 4). Furthermore, the intricate analysis on the distribution of exact, related and missed candidate matches are discussed in the section 4.1.3.

4.1.1 How much data do we need?

In this section, we discuss the candidate generation performance of our approaches trained using varying number of examples. It can be seen that the performance of miniEL has a negligible training impact and the performance is stable across different number of examples (see tables 2 and 3). However, the miniEL* approach improves consistently with increasing number of training examples.The miniEL approach without any finetuning still outperforms the miniEL* approach trained on all the training examples.

	miniEL*		miniEL	
Training Samples	R@1	R@5	R@1	R@5
0	0.401	0.594	0.553	0.756
10	0.427	0.622	0.552	0.758
1000	0.499	0.693	0.557	0.766
10000	0.518	0.717	0.553	0.76
ALL	0.534	0.736	0.556	0.756

Table 2: This table shows the R@1 and R@5 candidate generation performance of the approaches on the Medmentions (st21pv) dataset. The models are trained with varying number of training samples used to train/finetune the MiniEL* and MiniEL models.

In comparison, our approach outperforms generative methods like BioBART (Yuan et al., 2022a) and BioGenEL (Yuan et al., 2022b) that are resource intensive. Since these approaches use the Medmentions training set to finetune the models, we only compare the test set performance. The R@1 candidate generation performance of MiniEL is 0.552 as compared to the overall performance of 0.496 and 0.520 of BioBART and BioGenEL respectively (the results are taken from (Kartchner et al., 2023)).

	MiniEL*		MiniEL	
Training Samples	R@1	R@5	R@1	R@5
0	0.462	0.657	0.567	0.782
10	0.477	0.676	0.565	0.783
1000	0.525	0.728	0.569	0.789
10000	0.537	0.747	0.568	0.788
ALL	0.556	0.761	0.568	0.783

Table 3: This table shows the R@1 and R@5 candidate generation performance of the approaches on the Medmentions (full) dataset. The models are trained with varying number of training samples used to train/finetune the MiniEL* and MiniEL models.

4.1.2 Reranking Performance

In the following subsections, we discuss the candidate reranking results. The results corresponding to the parametric approach and those corresponding to the semantic disambiguation approaches are discussed in the following subsections.

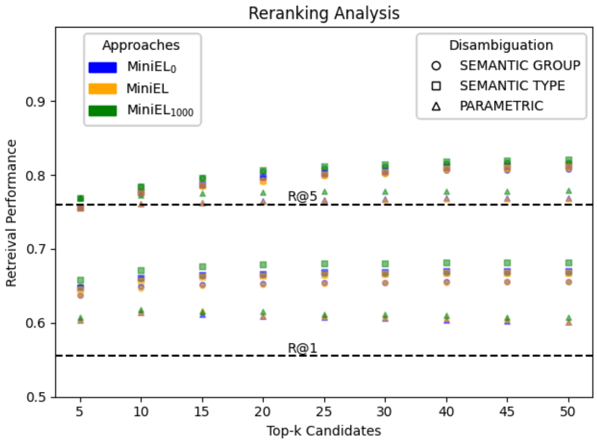


Figure 2: This figure highlights the trends associated with the retrieval performance improvement over varying top-k candidates using MiniEL$_0$, MinEL and MiniEL$_{1000}$ models. The improvement in R@1 is more significant as compared to that in R@5 for all the models and reranking methods. It can be observed that the retrieval performance of *PARAMETRIC* reranking decreases with increase in the top-k (k>15) whereas the performance of *SEMANTIC GROUP* and *SEMANTIC TYPE* reranking is consistent across the top-k.

1. **Parametric Reranking**: The top-k candidates selected based on the parametric approach discussed in section 3.3.1 and the corresponding results are shown in figure 2 and

	st21pv		full	
Reranking	top-5	top-10	top-5	top-10
PARAMETRIC	0.604	0.614	0.620	0.630
GROUP	0.638	0.649	0.659	0.670
TYPE	0.648	0.661	0.681	0.697

Table 4: The table shows the R@1 performance of the MiniEL$_0$ model after applying the listed reranking methods using the top-5 and top-10 candidates. It can be seen that there is a significant improvement in the performance as compared to the results in Tables 2 and 3.

Approach	Exact	Related	Missed
MiniEL$_0$	0.553	0.220	0.227
MiniEL$_0$ + PARAMETRIC	0.614	0.172	0.214
MiniEL$_0$ + GROUP	0.649	0.188	0.163
MiniEL$_0$ + TYPE	0.661	0.176	0.163

Table 5: This table shows the $R@1$ retrieval performance distributed into the exact matches, related matches and missed matches. The top-10 candidates are used for reranking. Here, we use the st21pv version of Medmentions.

table 4. It can be seen that the retrieval performance improves by from 0.553 to 0.614 for the st21pv version and from 0.567 to 0.630 for the full version of Medmentions. The a, b and c values used to obtain these results have the proportion $a : b : c \propto 50 : 2 : 1$ (see section A.2 for more details).

2. **With UMLS Semantic Information**: In this section, we discuss the retrieval performance improvements after the reranking using the semantic type and group information. The details of these methods are discussed in section 3.3.2.

   Figure 2 and table 4 show the R@1 performance of the MiniEL$_0$ model after applying these reranking strategies. The performance improves from 0.553 to 0.649 for semantic group and to 0.661 for semantic type reranking for the st21pv version of Medmentions. Similar observations can be made for the full version of Medmentions. Moreover, the retrieval performance does not deviate significantly with the increase in the top-k candidates used for reranking (see figure 2 for details).

The improvement in candidate ranking is approached in two ways. Firstly, to maximize the R@1 performance by reranking the generated candidates (see details in section 3.3) and secondly, to include context for addressing the context based ambiguity (see details in Appendix in section A.3).

4.1.3 How should the performance be evaluated?

In the retrieval-based evaluation strategy, we compute the retrieval performance on gold standard and predicted entity matches. However, there are cases where the most similar candidate is related to the gold standard entity. It can be seen in the table 5 that about 77% entities are exacting matching or are related to the gold standard entity. The details of each type of relation we have considered are provided by UMLS.[3]

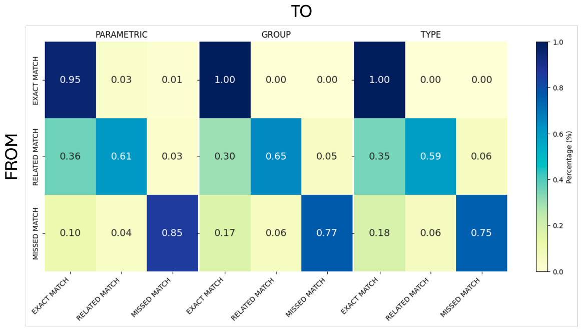


Figure 3: This heatmap illustrates the percentage changes in the number of initial exact, related and missed matches for the MiniEL$_0$ model. The performance preceding the changes is labeled 'FROM' for the rows, while the subsequent performance is denoted by 'TO' for the columns. The experiments are performed on the st21pv version of Medmentions.

Figure 3 shows that the effect of parametric reranking is directed primarily towards converting related matches to exact matches, coverting 36% of related matches into exact matches. The semantic group and semantic type based reranking approaches convert both missed and related matches into exact matches.

The following analysis is focused on the further evaluation of related and missed matches. In this article level analysis, we replace a mention with the closest generated candidate's canonical name for each mention in the article where the closest candidate is a related match or a missed match respectively. This results in an article A_P. We compute the cosine similarity between the original article A and the modified aricle A_P called S_P using a

[3] https://www.nlm.nih.gov/research/umls/knowledge_sources/metathesaurus/release/abbreviations.html#mrdoc_REL

PubmedBERT-base (Gu et al., 2020) model[4] fine-tuned using sentence transformers (Reimers and Gurevych, 2019) on biomedical data. Similarly, we also replace the mentions with the gold standard canonical names to create an article A_G. This is followed by computation of cosine similarity between A and A_G called S_G. We focus on scenarios where S_P and S_G deviate significantly as compared to the mean deviation of the articles. These are highlighed in the figure 4. This forms a base for our qualitative analysis where we use this deviation to provide insights on the granularity of gold standard predictions as well as highlight current issues in the approach.

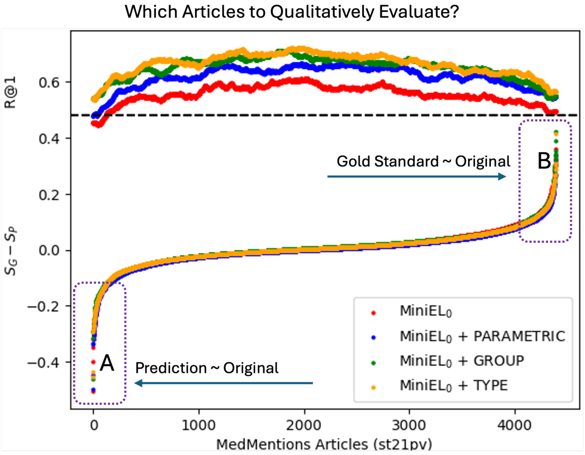


Figure 4: This figure illustrates the disparity in similarity scores ($S_G - S_P$) at the article level (4392 articles), alongside the smoothed retrieval performance (R@1) per article using a moving average with a window size of 200. The region A consists of semantically closer predictions and B consists of semantically farther predictions.

4.2 Qualitative Analysis

We perform a qualitative analysis on the entity linking predictions to highlight the difference in the granularity of the gold standard and predicted entities.

In this section, we qualitatively evaluate the articles displayed in the regions A and B of figure 4. The region A consists of articles where the predicted article A_P is semantically more similar to the original article A as compared to the gold standard articleA_G. Whereas, the region B consists of articles where A_G is more similar to A as compared to A_P.

MENTION: *"Vitamin D Receptor Activator Use and Cause-specific...Vitamin D receptor activators (VDRA) may exert...5,635 VDRA users were matched...that VDRA use was"* **GOLD**: Biologically Active Substance (C0574031) **PREDICTION**: VDR protein, human (C3657722) with parent entity Vitamin D3 Receptor (C0108082)
MENTION: *"Influence of Sinus Floor Configuration....the sinus floor configuration...osteotome sinus grafting procedure...into the sinus area...sinus floor configuration...sinus floor profile...flat sinus group...maxillary sinus following...predictable in sinuses with a concave..."* **GOLD**: Anatomical space structure (C0229984) **PREDICTION**: Nasal sinus (C0030471)
MENTION: *"...effectiveness of disc synoptoscope on patients...effectiveness of disc synoptoscope on binocularity...therapy with disc synoptoscope in...with disc synoptoscope is effective...disc synoptoscope could serve as an..."* **GOLD**: Medical Devices (C0025080) **PREDICTION**: Synoptophores (C0183765)
MENTION: *"...performance of the Afirma gene expression classifier...the Afirma gene expression classifier (GEC)...on which GEC was performed...GEC testing was performed...atypia of undetermined significance (AUS)...the AUS cases...the AUS group...patients with AUS...value of GEC decreased from...suspicious GEC result...value of GEC in indeterminate...suspicious GEC result...suspicious GEC result..."* **GOLD**: Research Activities (C0243095), Finding (C0242481) **PREDICTION**: Gene Expression Profiling (C0752248), Atypical cells of undetermined significance (C0522580)
MENTION: *"including the cytoplasmic tails of integrins and components of the actin cytoskeleton"* **GOLD**: CytoPlasmic (C0521449) **PREDICTION**: Cytoplasmic Domain (C1511625) with alias 'Cytoplasmic Tail'.

Table 6: The table shows qualitative examples selected from the region A in the figure 4.

Table 6 shows the qualitiative examples from region A where it can be observed that our approach is penalized for granular or highly related predictions. For example, The mention *gene expression classifier* has a gold standard entity *Research Activities* as compared to the more granular prediction *Gene Expression Profiling*. Similarly, the mention *cytoplasmic tails* has a gold standard entity *CytoPlasmic* as compared to the more granular prediction *Cytoplasmic Domain*.

Table 7 shows the qualitatuve examples corresponding to the region B where it can be seen that the gold standard annotation is based on the context of mention in the article. More specifically, the mention *mice* has a gold standard entity: *Laboratory mice* based on the article context. However, this context is missing in the mention surface form. Therefore, to address these kind of cases, we need to provide the necessary context in the query. We utilize three different disambiguation techniques and show examples of the corresponding predictions. We observe that additional context from the articles may result in granular predictions.

[4] https://huggingface.co/NeuML/pubmedbert-base-embeddings

However, the results are highly sensitive to the context and overall retrieval performance drops significantly (see section A.3 for more details).

We also observe an inconsistency in the granularity of gold standard entities in these examples. The mention *experimental mice* has a gold standard entity *Animals, Laboratory* as compared to the more granular prediction *Laboratory mice*.

MENTION: *"....iron accumulation in the substantia nigra (SN) of mice.....the substantia nigra of experimental mice treated with MPTP."*
GOLD: Laboratory mice (C0025929), Animals, Laboratory (C0003064)
PREDICTION: House mice (C0025914), Laboratory mice (C0025929)

MENTION: *"...mRNA N6-methyladenosine methylation of post-natal...mRNA m6A methylation during...outcomes of mRNA m6A methylation...levels of m6A methylation and...by m6A methylation at...higher m6A methylation and...differential m6A methylation may..."*
GOLD: mRNA methylation (C2611689)
PREDICTION: Methylation (C0025723)

Table 7: The table shows qualitative examples selected from the region B in the figure 4.

5 Conclusion

Biomedical entity linking has been an active area of research with various approaches being proposed to improve medical text standardization (see details in section 1). We propose a multi-stage approach where the first stage retrieves candidates with a high recall ($\sim 87\%$ for top-128 candidates). This is followed by application of the proposed reranking approaches focused on improving the R@1 retrieval performance. The reranking improves the performance by more than 10% (see figure 2 and table 4). We investigate the misses in R@1 and segregate the candidates into related and missed matches. Following this, we compute the article level semantic similarity together with the article level retrieval performance. This analysis highlights qualitative examples that can be used to obtain further insights about the framework. The semantic analysis is used to select the following types of qualitative examples: a) low retrieval performance and high similarity and, b) low retrieval performance and low similarity. The former can be highlight issues pertaining to granularity of gold standard entities and the latter can be used to highlight issues pertaining to the retrieval performance. Overall, the proposed techniques are highly effective in entity linking and have negligible training, prototype-space creation and inference costs (see table 9 for more details).

5.1 Future Scope

We believe that there is a significant scope for future developments in biomedical entity linking across different components of existing deep learning solutions. Firstly, there can be multiple biomedical normalizations for a mention or surface form. However, there is no method to determine the "closeness" of a prediction to a surface form as opposed to the binary matching. We believe that there should be a partial scoring instead of a binarized computation in order to accomodate the quality of predictions in the evaluation. Moreover, semantic similarities can also determined by experts to provide a ranking that could be used across biomedical entity linking for disambiguation.

5.2 Limitations

We observe that while an abbreviation preprocessing module is utilized in the proposed approaches, it doesn't convert all the abbreviations into their full forms. This causes a high amount of ambiguity in the results and often times the retrieval candidates do not consist of the correct entity. This drawback in positive pairs based learning has also been highlighted in (Zhang et al., 2021). Research addressed towards improving abbreviation expansion can help improve the recall of our candidate generation. Moreover, the region B in figure 4 highlights the examples where missing context in the surface form causes our framework to predict broader entities as the closest candidates. We utilize various approaches to include additional implicit and explicit context into our queries and analyze the corresponding retrieval performance (see details in Appendix section A.3).

References

Alan R Aronson. 2001. Effective mapping of biomedical text to the umls metathesaurus: the metamap program. In Proceedings of the AMIA Symposium, page 17. American Medical Informatics Association.

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. Nucleic acids research, 32(suppl_1):D267–D270.

Leo Breiman. 2001. Random forests. Machine learning, 45:5–32.

Nello Cristianini and John Shawe-Taylor. 2000. An introduction to support vector machines and other kernel-based learning methods. Cambridge university press.

Shaker El-Sappagh, Francesco Franda, Farman Ali, and Kyung-Sup Kwak. 2018. Snomed ct standard ontology based on the ontology for general medical science. BMC medical informatics and decision making, 18:1–19.

Samuele Garda and Ulf Leser. 2024. Belhd: Improving biomedical entity linking with homonoym disambiguation. arXiv preprint arXiv:2401.05125.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2020. Domain-specific language model pretraining for biomedical natural language processing.

Matthew Henderson, Rami Al-Rfou, Brian Strope, Yun-Hsuan Sung, László Lukács, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and Ray Kurzweil. 2017. Efficient natural language response suggestion for smart reply. arXiv preprint arXiv:1705.00652.

David Kartchner, Jennifer Deng, Shubham Lohiya, Tejasri Kopparthi, Prasanth Bathala, Daniel Domingo-Fernández, and Cassie Mitchell. 2023. A comprehensive evaluation of biomedical entity linking models. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 14462–14478, Singapore. Association for Computational Linguistics.

Luyang Kong, Christopher Winestock, and Parminder Bhatia. 2021. Zero-shot medical entity retrieval without annotation: Learning from rich knowledge graph semantics. arXiv preprint arXiv:2105.12682.

Linh Le, Guido Zuccon, Gianluca Demartini, Genghong Zhao, and Xia Zhang. 2022. Leveraging semantic type dependencies for clinical named entity recognition. In AMIA Annual Symposium Proceedings, volume 2022, page 662. American Medical Informatics Association.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics, 36(4):1234–1240.

Fangyu Liu, Ehsan Shareghi, Zaiqiao Meng, Marco Basaldella, and Nigel Collier. 2020. Self-alignment pretraining for biomedical entity representations. arXiv preprint arXiv:2010.11784.

Yuqing Mao, Randolph A Miller, Olivier Bodenreider, Vinh Nguyen, and Kin Wah Fung. 2023. Two complementary ai approaches for predicting umls semantic group assignment: heuristic reasoning and deep learning. Journal of the American Medical Informatics Association, 30(12):1887–1894.

Sunil Mohan and Donghui Li. 2019. Medmentions: A large biomedical corpus annotated with umls concepts. arXiv preprint arXiv:1902.09476.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. 2019. Scispacy: fast and robust models for biomedical natural language processing. arXiv preprint arXiv:1902.07669.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084.

Sunghwan Sohn, Donald C Comeau, Won Kim, and W John Wilbur. 2008. Abbreviation definition identification based on automatic precision estimates. BMC bioinformatics, 9:1–10.

Ali Soroush, Benjamin S Glicksberg, Eyal Zimlichman, Yiftach Barash, Robert Freeman, Alexander W Charney, Girish N Nadkarni, and Eyal Klang. 2024. Large language models are poor medical coders—benchmarking of medical code querying. NEJM AI, page AIdbp2300040.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. Advances in neural information processing systems, 30.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. Advances in Neural Information Processing Systems, 33:5776–5788.

Hongyi Yuan, Zheng Yuan, Ruyi Gan, Jiaxing Zhang, Yutao Xie, and Sheng Yu. 2022a. Biobart: Pretraining and evaluation of a biomedical generative language model. arXiv preprint arXiv:2204.03905.

Hongyi Yuan, Zheng Yuan, and Sheng Yu. 2022b. Generative biomedical entity linking via knowledge base-guided pre-training and synonyms-aware fine-tuning. arXiv preprint arXiv:2204.05164.

Sheng Zhang, Hao Cheng, Shikhar Vashishth, Cliff Wong, Jinfeng Xiao, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Knowledge-rich self-supervision for biomedical entity linking. arXiv preprint arXiv:2112.07887.

A Appendices

A.1 Terminology and Parameters

This section includes the terminology details and training, inference or other parameters used in this work.

Term	Description
δ	Similarity function
m	mention
ϵ	Entity
q	Query
μ	Entity canonical name
$TUI(.)$	maps an entity to it's semantic type canonical name
$SG(.)$	maps an entity to it's semantic group canonical name
$R@n$	Retrieval performance on top-n unique candidate entities
top-k	top-k candidate entities including aliases

Table 8: The table shows the symbols used in our work and the corresponding descriptions.

Table 9 shows the memory consumption and carbon emissions associated with the MiniEL$_0$ approach. It can be seen that our proposed techniques is low resource and results in very low amount of carbon emissions.

Phase	Memory (MB)	Emissions (Kg. Eq. CO2)
Training	0	0
Prototype Space Creation	1906	0.1
Inference	938	0.04

Table 9: The table shows the memory and carbon emission details. We utilized a 16GB V100 GPU for our tasks. The Inference was performed on the st21pv version of Medmentions.

A.2 Ablation Studies

In this section, we discuss the influence of parameters used in the parametric disambiguation approach discussed in the section 3.3. Specifically, we consider the candidate generation results obtained by using the MiniEL$_0$ model and perform the reranking by removing b and c parameters respectively. To highlight the impact of changing the a, b and c values, we perform a grid search on a manually selected range of values.

Furthermore, considering the top-10 candidates for reranking, removal of the parameter b results in an R@1 of 0.611, removal of c results in 0.481. This can be compared to the baseline R@1 0.553 and the R@1 of 0.614 obtained using optimal a,b and c. The performance is computed on the st21pv version of Medmentions. Overall, the impact of parameter c is highly significant in the performance improvement.

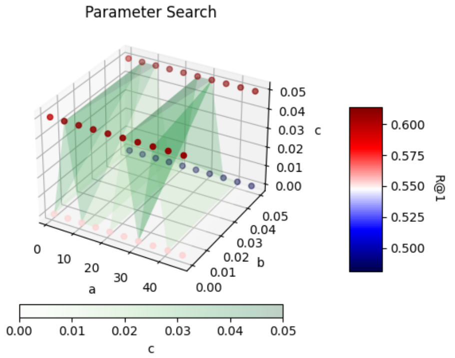


Figure 5: This figure shows the grid search on the parameters a, b and c for optimizing the R@1 performance of the MiniEL$_0$ model using the parametric approach discussed in the section 3.3. The optimal combination of a, b and c is found to be 5, 0.1 and 0.05, respectively.

A.3 Contextualized Queries

In our framework, the encoded representations of mentions are queried on the prototype space to get relevant candidates from UMLS. However, the mention spans alone may lack the necessary context to map the mention to their corresponding UMLS entities. In this section, we evaluate multiple techniques for incorporating context in the queries. Specifically, we use a running span based context addition, an implicit context addition and an attention based span context addition.

A.3.1 Neighboring Context

In this approach, we select a few words before and after the mention span to update the mention m and encode the updated mention to form a query.

Firstly, we add 5 neighbouring words before and after the mention and observe that the retrieval performance drops drastically ($R@1 \sim 10\%$). Therefore, we the number of words to 2 on both sides of the mention which results in a drastic drop in retrieval performance ($R@1 \sim 22\%$).

Overall, this context addition approach results in a significant drop in our retrieval performance and may not be suitable for contextual disambiguation.

A.3.2 Attention-Based Context

In this section, we perform experiments to identify the most influential words from the articles that attend to the span in consideration. We modify the original mentions by adding these words as additional context. This is done by utilizing the attention mechanism of encoder based transformer models namely BioBERT (Lee et al., 2020). Firstly, the entire title and abstract text is tokenized and passed to these models. The corresponding attention outputs are obtained and passed to the mention enrichment algorithm.

Let k be the number of word-piece tokens obtained from the encoder model, for each head H of Layer L, the attention matrix can be mentioned as:

$$A = \begin{bmatrix} a_{11} & a_{12} & ... & a_{1k} \\ a_{21} & a_{22} & ... & a_{2k} \\ \vdots & \vdots & .... & \vdots \\ \vdots & \vdots & .... & \vdots \\ a_{k1} & a_{k2} & .... & a_{kk} \end{bmatrix} \quad (5)$$

The mention spans lie in the range $[c, d]$ where $0 \leq c < d \leq k$. Therefore, the matrix A can be shortened to a submatrix of interest B mentioned as:

$$B = \begin{bmatrix} a_{cc} & a_{c(c+1)} & \cdots & a_{cd} \\ a_{(c+1)c} & a_{(c+1)(c+1)} & \cdots & a_{(c+1)d} \\ \vdots & \vdots & .... & \vdots \\ \vdots & \vdots & .... & \vdots \\ a_{kc} & a_{k(c+1)} & .... & a_{kd} \end{bmatrix} \quad (6)$$

Equivalently,

$$B = \begin{bmatrix} b_c & b_{(c+1)} & \cdots & b_d \end{bmatrix} \quad (7)$$

where b_i represents a column of B. Next, the token corresponding to the maximum attention value of each column is obtained as $T(max(b_i))$ where $T(j)$ represents the token at index $j \in \{1, 2, ..., k\}$in the text spanning from 1^{st} to the k^{th} token. The resulting token vector from the attention head H_m and Layer L_n is represented as:

$$R_{nm} = \begin{bmatrix} T(max(b_c) & T(max(b_{(c+1)}) & \cdots & T(max(b_d) \end{bmatrix} \quad (8)$$

The *ENRICH* function discussed in the algorithm 1 return the enriched context for a given mention m, which is then modified as shown below:

$$m* = m : R_{mn}[0], R_{mn}[1] \quad (9)$$

Finally, stop words are removed from $R_{mn}[0]$ and $R_{mn}[1]$. An example mention cold can be modified as *cold:* ***severe,recent*** where, 'severe, recent' is the added context.

Algorithm 1 Enrichment Context Selection

```
procedure SORT$_{mcbl}$($V$: 1D vector) {most common by length in descending order}
    $C = \{x \mid count(x) = max(count(T)) \; \forall \; T \in V\}$
    $C^* = \{x \mid x \in C \; and \; len(x) >= len(y) \; \forall \; y \in C\}$
    return $C^*$
end procedure
{$R_n$ denotes the representative token from all attention heads in Layer n}
{$L_n$ denotes the representative token(s) from Layer n}
{$M$ denotes the representative token(s) for the token$_k$ in mention M}
{$E$ denotes the representative token context (E) for mention M}
procedure ENRICH($R_n$ : 1D vector) {enrich mention with context}
    $C^* = SORT_{mcbl}(R_{nm})$
    $R_n = C_1^*$ or $R_n = C^*(1)$
    $L_n = \{R_1, R_2, ..., R_z\}$
    $C^* = SORT_{mcbl}(L_n)$
    $M_t = \{C_1^*, C_2^*\}$
    $M = \{M_1, M_2, ..., M_k\}$
    $C^* = SORT_{mcbl}(M)$
    $E = \{C_1^*, C_2^*\}$
end procedure
```

A.3.3 Implicit Context

In this approach, we utilize mean-pooled embedding of the mention encodings taken from the entire article as an input. Firstly, the entire text is used as an input to obtain the tokenwise encodings from the model.

$$f(text, \theta) = \{E_{T_1}, E_{T_2}, ..., E_{T_n}\} \quad (10)$$

Here, E_T is encoding of token T and n are the number of tokens in the input text.

Given a span s, consisting of l tokens and tokens in the span $\{T_k, ..., T_{k+l}\}$, we take the corresponding encodings from the model outputs $\{E_{T_k}, ..., E_{T_{k+l}}\}$. We perform a mean pooling on these encodings to obtain the updated query representation $Q = \frac{1}{l}\sum_{k}^{k+l}\{E_{T_k}, ..., E_{T_{k+l}}\}$. The prototype space consists of the sentence encodings of the canonical names of all the entities in UMLS.

The R@1 candidate generation performance drops drastically in this setup where a drop of more than 30% is observed. Overall, we observe that these implicit contextual queries are not helpful in improvement of retrieval performance.

A.3.4 Evaluation

In this section, we perform the quantitative and qualitative analysis of our context based approaches on the Medmentions st21pv version. The qualitative examples shown below highlight the predictions provided by the proposed context based

MENTION: *"....iron accumulation in the substantia nigra (SN) of mice...."*
MENTION^{AC}: *"...iron accumulation in the substantia nigra (SN) of mice: experiment...."*
PREDICTION^{AC}: Laboratory mice (C0025929))
PREDICTION^{IC}: House mice (C0025914)
PREDICTION^{NC}: Laboratory mice (C0025929)

MENTION: *Kindlin-1 is expressed primarily in epithelial cells, kindlin-2 is widely distributed and is particularly abundant in adherent cells, and kindlin-3 is expressed primarily in hematopoietic cells.*
MENTION^{AC}: *Kindlin-1: kind,primarily is expressed primarily in epithelial cells, kindlin-2: distributed,Kind is widely distributed and is particularly abundant in adherent cells, and kindlin-3: expressed,Kind is expressed primarily in hematopoietic cells.*
PREDICTION^{AC,IC,NC}: FERMT1 gene (C1423809), FERMT2 gene (C1423716), FERMT3 protein, human (C1311640)
PREDICTION^{AC} **+ TYPE**: Fermitin Family Homolog 2, human (C3889282), Fermitin Family Homolog 2, human (C3889282), FERMT3 protein, human (C1311640)

Table 10: This table shows the qualitative analysis of the MiniEL$_0^{AC}$, MiniEL$_0^{IC}$ and MiniEL$_0^{NC}$ approaches on examples from Medmentions.

MENTION:"...inhibitor of T cell function....hypoxic conditions influence human T cell functions and found that..."
MENTION^{AC}:"...inhibitor of T cell function: cell....hypoxic conditions influence human T cell functions: cell and found that..."
GOLD: Cell physiology (C0007613), Cell physiology (C0007613)
PREDICTION^{AC}: Cell physiology (C0007613), Cell physiology (C0007613)
PREDICTION^{NC}: Cell physiology (C0007613), T cell differentiation (C1155013)

Table 11: This table shows the qualitative analysis of the MiniEL$_0^{AC}$ and MiniEL$_0^{NC}$ approaches on examples from Medmentions.

Approach	R@1	R@5
miniEL$_0$	0.553	0.756
miniEL$_0^{NC}$	0.219	0.405
miniEL$_0^{AC}$	0.384	0.642
miniEL$_0^{IC}$	0.161	0.359

Table 12: The table presents the candidate generation performance of the listed context based approaches. The performance is computed the st21pv version of Medmentions.

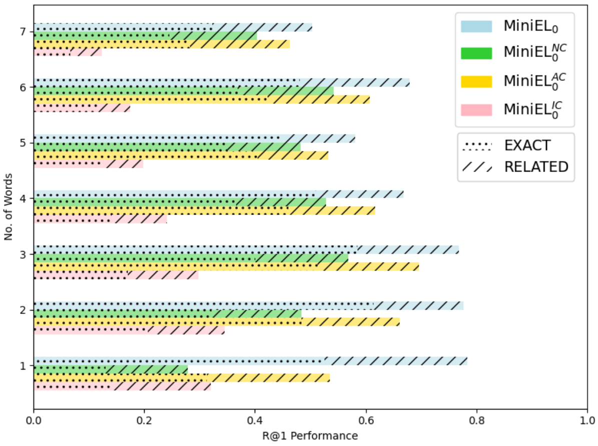


Figure 6: This figure presents the word-count level retrieval performance, measured in terms of exact and related matches, comparing the performance of the MiniEL$_0$ approach in comparison to its performance on applying the context based methods.

approaches. As discussed in the qualitative analysis of region B (see section 4.2), the surface forms have missing context resulting in an inaccurate prediction.

It can be observed in table 10 that the mention *mice* is correctly predicted as the entity *Laboratory mice* using the MiniEL$_0^{AC}$ and MiniEL$_{NC}$ reranking approaches. We also highlight the effect semantic type reranking approach though the example mentions *kindlin-2* and *kindlin-3* where the prediction semantic type changed from 'Gene' to the correct type 'Protein'. Here, the MiniEL$_0^{NC}$, MiniEL$_0^{AC}$ and MiniEL$_0^{IC}$ methods correspond to the results obtained using the Neighboring Context, Attention-based Context and Implicit Context approaches, respectively, utilizing MiniEL$_0$ as the base model.

It can be observed that the AC approach provides meaningful outputs as it includes the necessary context in the surface form. Similar outputs are provided by the NC approach. However, the neighbouring words may not necessarily contain the context and this can be seen in the following qualitative example listed in the table 11.

We observe that the attention span based context enrichment approach is sensitive to the context addition as it induces bias the surface form and the resulting candidates may be more similar to the bias term as compared to the base form. Therefore, to understand the impact of bias on the surface form, we observe the retrieval performance based on the number of words in the mention. The figure 6 shows that the performance of MiniEL$_0^{AC}$ is better on mentions with higher length as compared to the mentions with lower lengths. A similar trend is observed for the MiniEL$_0^{NC}$ approach. This trend is not seen for the MiniEL$_0^{IC}$ approach where the performance drops with the increase in number of words in the mentions. However, the attention span based approach has better performance as compared to the neighboring context approach. For each specific mention word count, we select mentions with at least about a 100 examples for this analysis.

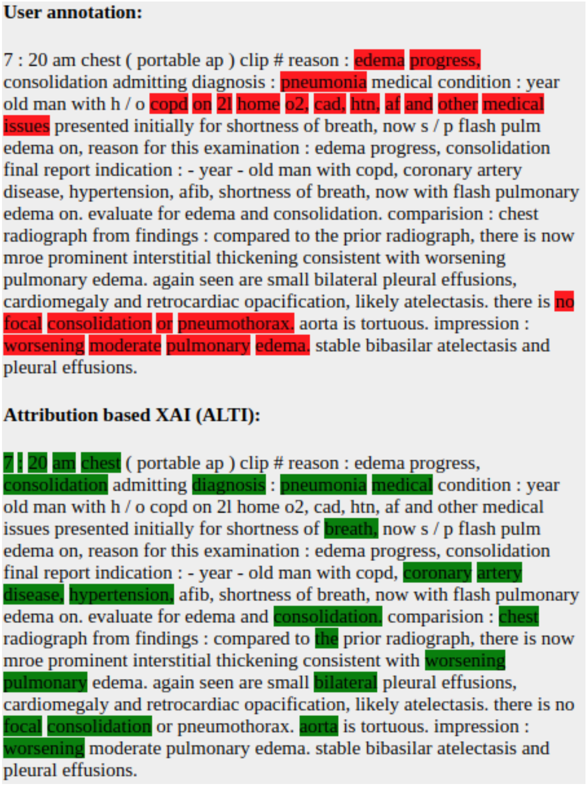
To summarize, the quantitative and qualitative context enrichment analysis shows that the MiniEL$_0^{AC}$ approach outperforms the other approaches and is effective in context addition. However, the sensitivity in the encodings results in large deviations in the candidate generation (see table 12). Therefore, the robustness of this contextual approach needs to be improved.

XAI for Better Exploitation of Text in Medical Decision Support

Ajay Madhavan Ravichandran[1] **Julianna Grune**[1] **Nils Feldhus**[1]
Aljoscha Burchardt[1] **Sebastian Möller**[1,2] **Roland Roller**[1]
[1]German Research Center for Artificial Intelligence (DFKI)
[2]Technische Universität Berlin
{firstname.lastname}@dfki.de

Abstract

In electronic health records, text data is considered a valuable resource as it complements a medical history and may contain information that cannot be easily included in tables. But why does the inclusion of clinical texts as additional input into multimodal models, not always significantly improve the performance of medical decision-support systems? Explainable AI (XAI) might provide the answer. We examine which information in text and structured data influences the performance of models in the context of multimodal decision support for biomedical tasks. Using data from an intensive care unit and targeting a mortality prediction task, we compare information that has been considered relevant by XAI methods to the opinion of a physician.



User annotation:

7 : 20 am chest (portable ap) clip # reason : edema progress, consolidation admitting diagnosis : pneumonia medical condition : year old man with h / o copd on 2l home o2, cad, htn, af and other medical issues presented initially for shortness of breath, now s / p flash pulm edema on, reason for this examination : edema progress, consolidation final report indication : - year - old man with copd, coronary artery disease, hypertension, afib, shortness of breath, now with flash pulmonary edema on. evaluate for edema and consolidation. comparision : chest radiograph from findings : compared to the prior radiograph, there is now mroe prominent interstitial thickening consistent with worsening pulmonary edema. again seen are small bilateral pleural effusions, cardiomegaly and retrocardiac opacification, likely atelectasis. there is no focal consolidation or pneumothorax. aorta is tortuous. impression : worsening moderate pulmonary edema. stable bibasilar atelectasis and pleural effusions.

Attribution based XAI (ALTI):

7 : 20 am chest (portable ap) clip # reason : edema progress, consolidation admitting diagnosis : pneumonia medical condition : year old man with h / o copd on 2l home o2, cad, htn, af and other medical issues presented initially for shortness of breath, now s / p flash pulm edema on, reason for this examination : edema progress, consolidation final report indication : - year - old man with copd, coronary artery disease, hypertension, afib, shortness of breath, now with flash pulmonary edema on. evaluate for edema and consolidation. comparision : chest radiograph from findings : compared to the prior radiograph, there is now mroe prominent interstitial thickening consistent with worsening pulmonary edema. again seen are small bilateral pleural effusions, cardiomegaly and retrocardiac opacification, likely atelectasis. there is no focal consolidation or pneumothorax. aorta is tortuous. impression : worsening moderate pulmonary edema. stable bibasilar atelectasis and pleural effusions.

Figure 1: Comparison of human annotation regarding relevant tokens for in-hospital mortality, versus XAI

1 Introduction

Electronic health records often contain factual information in short, tabular form, including laboratory values, diagnoses, gender, and age. They also include longer texts in various forms written for many different purposes. Depending on the origin and context of the data, the text could be a clinical or nursing note, a discharge summary, or a radiology report, to name a few. The text might provide a high-level interpretation of the current patient situation, taking different kinds and sources of information into account. The text might refer directly to some given structured facts in the database (e.g., a lab value is above borderline) but might also consider additional information such as general impressions, assumptions, and information gathered directly from the patient or other medical personnel (e.g., the patient is not very adherent). For this reason, the texts are generally considered valuable resources in the clinical routine. In the context of machine learning for healthcare, however, the inclusion of such texts has shown in various setups only marginal effects (Khadanga et al., 2019; Yang and Wu, 2021), although one would assume that the additional information and complementary perspective should improve a system's performance.

Many papers in this area deal with multi-modal data, integrating, for instance, image and text, or structured and unstructured data into one model. MIMIC-III (Johnson et al., 2016) is a popular dataset in this context, as it can be easily accessed by researchers. It contains data from an intensive care unit (ICU) of a US hospital, including patient demographics, time series data, or text, such as nursing notes, discharge summaries, or social worker notes. However, while many approaches in other domains do achieve a boost in performance using multimodal (text) data, the performance dif-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 506–513
August 16, 2024. ©2024 Association for Computational Linguistics

ference between unimodal and multi-modal models in the medical context can be modest (Deznabi et al., 2021). In this work, we explore which information is valuable for multi-modal machine learning using MIMIC data. More precisely, we re-implement two multi-model (MM) approaches for the task of in-hospital mortality prediction. We then introduce an XAI approach for the given MM approaches and examine the attributed information according to their faithfulness. Finally, we investigate if the attributions are plausible from a physician's perspective.

2 Related Work

Recent years have seen a surge in leveraging deep learning approaches utilizing diverse clinical data sources for clinical outcome predictions. These include textual clinical notes, longitudinal data, and demographic data. Unimodal approaches like CNNs (Rocheteau et al., 2021), LSTMs (Choi et al., 2016), and BERT (Naik et al., 2022) have laid the groundwork. Later expanded to multimodal approaches such as additive fusion (Khadanga et al., 2019; Deznabi et al., 2021) to more sophisticated cross attention fusion (Zhang et al., 2022; Qiao et al., 2019). Yang and Wu (2021) and Deznabi et al. (2021) implemented additive and gated fusion-based multimodal models for tasks like diagnosis prediction, acute respiratory failure prediction, and in-hospital mortality prediction. We extend their work by applying explainability methods to models and evaluating the quality of explanations.

Explainable AI (XAI) enhances transparency and trust in healthcare applications, especially within medical decision support systems (Markus et al., 2021) and clinical NLP (Roller et al., 2022a). Notably, Naylor et al. (2021) compared the faithfulness of various explanation methods for models like BERT in mortality prediction. Additionally, DeYoung et al. (2020) introduced a benchmark with human annotations to evaluate NLP models explainability for faithfulness and plausibility. However, previous research has mainly focused on quantitative evaluations of explainability methods for uni models. This study addresses this gap by quantitatively evaluating XAI in multimodal models.

3 Method

3.1 Data

We use the Medical Information Mart for Intensive Care (MIMIC-III) dataset (Johnson et al., 2016) in our experiments. MIMIC comprises authentic electronic health record (EHR) data, including vital signs, laboratory measurements, and clinical notes (free text), from ICU patients. One of its tasks involves predicting patient mortality risk in the intensive care unit (ICU) based on the first 48 hours of patient stay. Mortality, in this context, refers to the likelihood of a patient dying while receiving intensive care.

For our cohort selection and setup, we mostly follow Harutyunyan et al. (2019) and Yang and Wu (2021) and focus on patients aged 18 years and older with ICU stays lasting 48 hours or more, accompanied by clinical notes. The original cohort of Harutyunyan et al. (2019) includes 17 different features that undergo different pre-processing steps, such as inserting missing information by previous or plausible default values and converting them into time series data. As we are particularly interested in text data, we extend the original cohort by two different sources of text, namely nursing notes and admission notes.[1]

The final data consists of three different modalities: a) text, consisting of either nursing notes or admission notes; b) time series data, such as heart rate, blood pressure, or glucose; and c) time-invariant data, such as age or ethnicity. While some time series features are numeric, others are categorical (e.g., Glasgow coma scale eye-opening), which are converted into several binary features during pre-processing following the approach of Harutyunyan et al. (2019). More details about data imputation and a synthetic example of the data are added to the Appendix B.

3.2 Multimodal Models

In this study, we employ diverse architectures to encode information from different modalities into latent vectors. Specifically, we use LSTMs to process time series data, linear layers to handle time-invariant data, and transformer models for textual data. To integrate all the encoded information effectively, we use two fusion approaches: The gated fusion approach proposed by Yang and Wu (2021) and the concatenation fusion approach introduced by Deznabi et al. (2021). In the gated fusion approach, a gated attention mechanism is applied over the encoded vectors to generate a fused representation that incorporates context from all the encoded

[1] Explanation of this terminology can be found in Appendix A.

vectors. Conversely, in the concatenated fusion approach, all the encoded vectors are concatenated into a single vector to produce a fused representation. Figure 2 depicts a simplified overview of the multimodal architectures. Subsequently, the fused vector from both fusion approaches is projected into a fully connected layer for prediction.

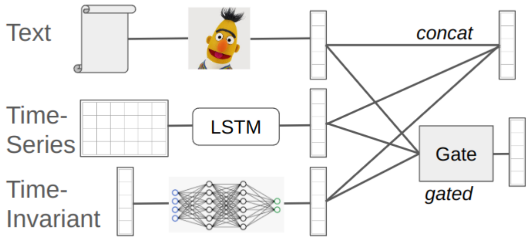


Figure 2: Combining modalities using concatenation or gated fusion.

Both approaches use a pre-trained ClinicalBERT (Huang et al., 2019) to encode the clinical notes (nursing, radiology, others, etc.). Both approaches use the average embedding over all the clinical notes as the encoded textual feature.

3.3 Multi-Modal XAI

To identify which information is crucial for successful predictions in our multimodal setup, we integrate XAI techniques using state-of-the-art methods based on gradient and attention. For pinpointing significant information in time series data processed by the LSTM, we employ Integrated Gradients (IG) (Sundararajan et al., 2017). For the textual data fed into the BERT model, we use the attention vector norm (Kobayashi et al., 2020) and layer-wise Token-to-token Interaction (ALTI) (Ferrando et al., 2022). These methods have shown promising results in explaining transformer-based models. They let us identify relevant tokens in the texts and pertinent features in the time series data, which we can then compare with annotations provided by medical professionals.

4 Experimental Results

Our first experiment concerns the reproduction of multimodal and unimodal methods and application to the in-hospital mortality task. For the evaluation, we follow a similar methodology to related work (§2), utilizing ROC (Area Under the Receiver Operating Characteristic curve) and AUPR (Area Under the Precision-Recall curve).

In the second experiment, we apply XAI to the models and examine which information is considered by the model as valuable for the prediction. Following Jacovi and Goldberg (2020), we explore faithfulness by replacing the top X attributed token or time point of the time series with a mask token or zero and observe the drop in model performance.

Finally, we conduct a plausibility test, as suggested by DeYoung et al. (2020). Here, we directly compare the attributions on text and structured data to the relevant information according to a physician's perspective. Only annotation of text data is quantitatively analyzed based on the overlap between annotated tokens and attributed tokens, such overlap matching is not possible for time-series data. As we are particularly interested in examining the benefit of text data, we randomly select 100 patient cases in which a multi-modal approach predicts a higher probability score for mortality than the unimodal LSTM approach. Likewise, we randomly select 100 cases in which the multi-modal approach predicts a lower probability score for mortality. For those cases, we assume that text data provided additional information to make a stronger prediction assumption (independent of whether the prediction is correct or not).

A final-year medical student annotated these 200 cases. The student was asked to identify parts of the text and important time-series features that support the outcome of mortality or survival. In addition, the student was asked to provide their estimation of the patient's survival and whether the text was useful in solving the task.

4.1 Results

Model performance: Unimodal vs. multi-modal Table 1 presents the results of the two multimodal approaches in comparison to the unimodal models for both text types. The first observation is that LSTM provides stronger results compared to the two BERT approaches, and all multimodal approaches outperform the unimodal models. This slight performance gain is particularly visible when using nursing notes in comparison to admission notes. Moreover, the more complex gated mechanism shows a slight benefit over the concatenation. Overall, the presented results are comparable to what has been reported already in other related work (Khadanga et al., 2019; Lyu et al., 2022).

We can conclude that for the given data and the given problem, structured (time series) data seems to have a stronger influence on the model performance, and adding both 'worlds' can lead to further, but rather minor, improvements. However, an ad-

Table 1: Performance of multimodal (MM) approaches in comparison to the unimodal models in predicting in-hospital mortality according to ROC and AUPR.

Model	ROC	AUPR
BERT (nursing)	0.80	0.37
BERT (admission)	0.74	0.30
LSTM	0.80	0.42
ConcatMM (nursing)	0.81	**0.44**
ConcatMM (admission)	0.81	0.37
GatedMM (nursing)	**0.83**	0.43
GatedMM (admission)	0.82	0.39
LSTM w/o height+weight	0.78	0.38

ditional analysis of the data reveals that the two features *height* and *weight* are often missing and imputed with default values. For this reason, we removed those two features from the original data and trained an LSTM. Without those two features, however, the model suffers a drop in performance.

Explanation faithfulness test Table 2 presents the faithfulness test, in which we examine which information influences the models' prediction. To do so, we replace the top-5 (top-10 and top-15) strongest (XAI) attributed tokens or time points of the time-series data and compare this to a random replacement of the same amount of information. The table shows that removing the attributed tokens leads to a stronger drop in performance, compared to the random removal. This indicates that the model relies on information (and particularly text tokens), which are useful for the mortality prediction task.

Table 2: ROC Performance after replacement of top-X text tokens or time point of time-series data. The table compares a random replacement against the replacement of attributed information (XAI). The table compares BERT (admission) with ROC=0.80 for text and MM with ROC=0.83.

Modality	Top	Attribution	Random
Text	5	0.769 (0.031)	0.801 (0.000)
	10	0.744 (0.056)	0.800 (0.000)
	15	0.734 (0.066)	0.799 (0.001)
Struct.	5	**0.664** (0.166)	0.726 (0.104)
	10	**0.595** (0.235)	0.674 (0.156)
	15	**0.585** (0.245)	0.632 (0.198)

Regarding the attributed tokens in the text data, we found the following patterns: First, highly attributed information is often spread widely across the document. In many cases, attributed tokens in a document include medical conditions such as symptoms or diseases (e.g., pain, cirrhosis, pneumonia), in some others also body parts such as heart or lung and sometimes medications. However, many other seemingly irrelevant tokens are highlighted, such as the word *patient* or a specific time mentioned in the text. Finally, even though information tends to be spread across the document, the attribution also covers sequences of words, such as the patient's age ('53 y. o. man'), negations ('denies pain'), and other connected information ('chest pain,' 'renal failure').

When looking closer at the attributed time-series data, the following five features play a particularly important role in the model's performance drop: Glasgow Coma Scale (total), blood pressure (mean), Glasgow Coma Scale (motor response), oxygen saturation, and Glasgow Coma Scale (verbal response).

Explanation plausibility evaluation For the 200 patient cases that a physician annotated, we first conducted a manual analysis to find differences and similarities to the attributed tokens. Figure 1 depicts an example text with human and machine (XAI) annotation. In general, the annotations show that, in many cases, a few larger chunks of text sequences were annotated. Moreover, even though severe conditions seem to be mentioned multiple times in the documents (redundancy), the physician often annotated each condition just once – the explanations, however, also highlight the same condition in multiple parts of the document. Moreover, the physician annotated some measurements of values as relevant, whereas XAI never detected anything comparable – although it considers, in some cases, age and gender as useful. On a time-series data, the physician considers similar information useful compared to XAI, namely the Glasgow Coma Scale (eye-opening), the Glasgow Coma Scale (motor response), the Glasgow Coma Scale (total), oxygen saturation, and respiratory rate.

Table 3: Plausibility evaluation measuring agreement with human-annotated of the clinical text (nursing and admission) for mortality prediction. The table shows the lenient-f1 scores obtained by measuring the overlap between the annotated token and the attributed token.

Model	Precision	Recall	Lenient-F1
BERT (nursing)	**0.141**	**0.204**	**0.166**
BERT (admission)	0.064	0.090	0.075
ConcatMM (nursing)	0.102	0.159	0.124
GatedMM (admission)	0.110	0.168	0.133

Second, we quantitatively evaluated plausibility by measuring the lenient-F1 score for the overlap between annotated and attributed text. Since our main focus is textual data, we did not create annotations for time points, making such overlap evaluation impossible for time series data. Table 3 shows that the BERT model attributions align more closely with human annotations for nursing notes, while multimodal models exhibit lower agreement with human annotations. However, the overall agreement, as measured by the lenient-F1 score, is very low. This low agreement is likely because the models struggle to differentiate between acute conditions (e.g., active bleeding, signs of severe infection) and pre-existing conditions (e.g., pneumonia, diabetes mellitus), missing out on the negation of medical conditions by attending only to pathology (e.g., 'no melena' is annotated by physician but the model attribution identifies only "melena").

5 Discussion

The initial results align with findings from related work: text data is a valuable resource for improving predictions, but its benefit varies depending on the task and the text source. For instance, nursing notes led to higher results than admission notes, despite the fact that nursing notes were often truncated due to BERT's restricted input length. Given the redundancy in clinical texts, it may be beneficial to compress larger texts into shorter documents to accommodate additional text sources.

Another notable finding is the performance drop when removing height and weight, two features that are often missing and filled with default values. Our medical expert confirmed that *height* and *weight* do not influence the given task, which may reduce overall trust in our model. However, it is not unusual for machine learning models to consider seemingly irrelevant information as useful. For example, in Roller et al. (2022b), a nephrology outcome prediction model found the number of lab measurements in the last month to be very useful, which may indirectly indicate a patient's deteriorating condition. In our case, the model's reliance on height and weight might be justified by the context in which these features are used. For instance, weight may be measured over time to monitor fluid balance. Thus, the model might be capturing an important dependency that is not immediately apparent.

In the second experiment evaluating the faithfulness of the attributions, we observed a significant drop in model performance when the top-attributed information was replaced in the input, compared to a random replacement. This stronger decline in performance was particularly pronounced when time-series data was replaced, indicating that time-series information plays a crucial role in the model's performance for the given task. Conversely, it also shows that some tokens, such as medical conditions that are mentioned in the text, have a positive influence on the model.

In the third experiment, comparing human and XAI annotations of texts suggests that systems can extract relevant information (pre-existing conditions are identified more often than acute conditions). On the other hand, the extracted information is not always humanly plausible. The comparison of human and XAI annotated time-series features showed that both the physician and the model consider similar features useful for the given prediction task. However, multimodal quantitative analysis of plausibility remains a bottleneck that should be addressed in future work.

6 Conclusion

We analyzed the relevance of text and structured data in the context of a multimodal decision support system for in-hospital mortality. We found that the source of text influences the model performance (nursing vs admission notes). Moreover, sparse information (e.g., patient height and weight) can benefit the performance of models, although such information does seem irrelevant from an expert's perspective.

In our experiments, we found that the model performance drops considerably when structured information (time series) is replaced in the input compared to textual inputs. In general text data could provide additional context in a multimodal setup, but its benefit depends on the task (other tasks might lead to more benefits) as our results showed only a marginal boost in performance compared to unimodal models.

Finally, our comparison between human and XAI annotations of the texts indicates that the models can extract relevant information but not always. It seems that for multimodal data such as text and time series, quantitative analysis of plausibility is a bottleneck, and it should be addressed in future work.

Limitations

Our approach has clear limitations in terms of applied models (for instance, a multimodal LLM could have been tested) as well as additional XAI methods (e.g. LIME or SHAP). Moreover, in order to gain more insights into the human perspective, a large-scale annotation from a human perspective is necessary, considering additional human annotators, patient cases, and datasets.

Ethical Considerations

Although we build multimodal machine learning models for healthcare with the intention of creating a positive impact on society, our model is trained and tested only on retrospective and anonymized data. In this way, we do not influence patient outcomes.

Acknowledgments

The project has received funding from the German Federal Ministry of Education and Research (BMBF) through the project PRIMA-AI (01GP2202C).

References

Edward Choi, Mohammad Taha Bahadori, Andy Schuetz, Walter F. Stewart, and Jimeng Sun. 2016. Doctor ai: Predicting clinical events via recurrent neural networks. In *Proceedings of the 1st Machine Learning for Healthcare Conference*, volume 56 of *Proceedings of Machine Learning Research*, pages 301–318, Northeastern University, Boston, MA, USA. PMLR.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2020. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online. Association for Computational Linguistics.

Iman Deznabi, Mohit Iyyer, and Madalina Fiterau. 2021. Predicting in-hospital mortality by combining clinical notes with time-series data. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4026–4031, Online. Association for Computational Linguistics.

Javier Ferrando, Gerard I. Gállego, and Marta R. Costa-jussà. 2022. Measuring the mixing of contextual information in the transformer. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8698–8714, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hrayr Harutyunyan, Hrant Khachatrian, David C. Kale, Greg Ver Steeg, and Aram Galstyan. 2019. Multitask learning and benchmarking with clinical time series data. *Scientific Data*, 6(1):96.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *arXiv preprint arXiv:1904.05342*.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, Online. Association for Computational Linguistics.

Alistair E W Johnson, Tom J Pollard, Lu Shen, Li-Wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

Swaraj Khadanga, Karan Aggarwal, Shafiq Joty, and Jaideep Srivastava. 2019. Using clinical notes with time series data for ICU management. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6432–6437, Hong Kong, China. Association for Computational Linguistics.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2020. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7057–7075, Online. Association for Computational Linguistics.

Weimin Lyu, Xinyu Dong, Rachel Wong, Songzhu Zheng, Kayley Abell-Hart, Fusheng Wang, and Chao Chen. 2022. A multimodal transformer: Fusing clinical notes with structured ehr data for interpretable in-hospital mortality prediction. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, 2022:719–728.

Aniek F. Markus, Jan A. Kors, and Peter R. Rijnbeek. 2021. The role of explainability in creating trustworthy artificial intelligence for health care: A comprehensive survey of the terminology, design choices, and evaluation strategies. *Journal of Biomedical Informatics*, 113:103655.

Aakanksha Naik, Sravanthi Parasa, Sergey Feldman, Lucy Lu Wang, and Tom Hope. 2022. Literature-augmented clinical outcome prediction. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 438–453, Seattle, United States. Association for Computational Linguistics.

Mitchell Naylor, Christi French, Samantha Terker, and Uday Kamath. 2021. Quantifying explainability in NLP and analyzing algorithms for performance-explainability tradeoff. *Interpretable ML in Healthcare workshop at ICML 2021*.

Zhi Qiao, X. Wu, Shen Ge, and Wei Fan. 2019. Mnn: Multimodal attentional neural networks for diagnosis prediction. In *International Joint Conference on Artificial Intelligence*.

Emma Rocheteau, Pietro Liò, and Stephanie Hyland. 2021. Temporal pointwise convolutional networks for length of stay prediction in the intensive care unit. In *Proceedings of the Conference on Health, Inference, and Learning*, CHIL '21, page 58–68, New York, NY, USA. Association for Computing Machinery.

Roland Roller, Aljoscha Burchardt, Nils Feldhus, Laura Seiffe, Klemens Budde, Simon Ronicke, and Bilgin Osmanodja. 2022a. An annotated corpus of textual explanations for clinical decision support. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2317–2326, Marseille, France. European Language Resources Association.

Roland Roller, Manuel Mayrdorfer, Wiebke Duettmann, Marcel G Naik, Danilo Schmidt, Fabian Halleck, Patrik Hummel, Aljoscha Burchardt, Sebastian Möller, Peter Dabrock, Bilgin Osmanodja, and Klemens Budde. 2022b. Evaluation of a clinical decision support system for detection of patients at risk after kidney transplantation. *Frontiers in Public Health*, 10.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR.

Betty van Aken, Jens-Michalis Papaioannou, Manuel Mayrdorfer, Klemens Budde, Felix A. Gers, and Alexander Löser. 2021. Clinical outcome prediction from admission notes using self-supervised knowledge integration. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 881–893. Association for Computational Linguistics.

Bo Yang and Lijun Wu. 2021. How to leverage the multimodal EHR data for better medical prediction? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4029–4038, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ying Zhang, Baohang Zhou, Kehui Song, Xuhui Sui, Guoqing Zhao, Ning Jiang, and Xiaojie Yuan. 2022. PM^{2}F^{2}N: Patient multi-view multi-modal feature fusion networks for clinical outcome prediction. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1985–1994, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

A Nursing and admission notes

Nursing notes are small text snippets written by medical personnel during a patient's stay in the ICU. They describe general observations, current medical conditions, and treatment. As we target the patient situation within the first 48 hours, we concatenate all nursing notes from that time into one document.

Following the work of van Aken et al. (2021), we simulate patient textual information at the time of admission by extracting the chief complaint, present illness, medications, allergies, physical exam, and family and social history from discharge summaries. We refer to this as admission notes.

B Imputation value and synthetic sample

Table 4: Shows the selected time-series values and their corresponding impute values (plausible).

Variable	**Impute value**
Capillary refill rate	0
Diastolic blood pressure	59.0
Fraction inspired oxygen	0.21
Glascow coma scale eye opening	4 spontaneously
Glascow coma scale motor response	6 obeys commands
Glascow coma scale total	15
Glascow coma scale verbal response	5 oriented
Glucose	128.0
Heart Rate	86
Height	170.0
Mean blood pressure	77.0
Oxygen saturation	98.0
Respiratory rate	19
Systolic blood pressure	118.0
Temperature	36.6
Weight	81.0
pH	7.4

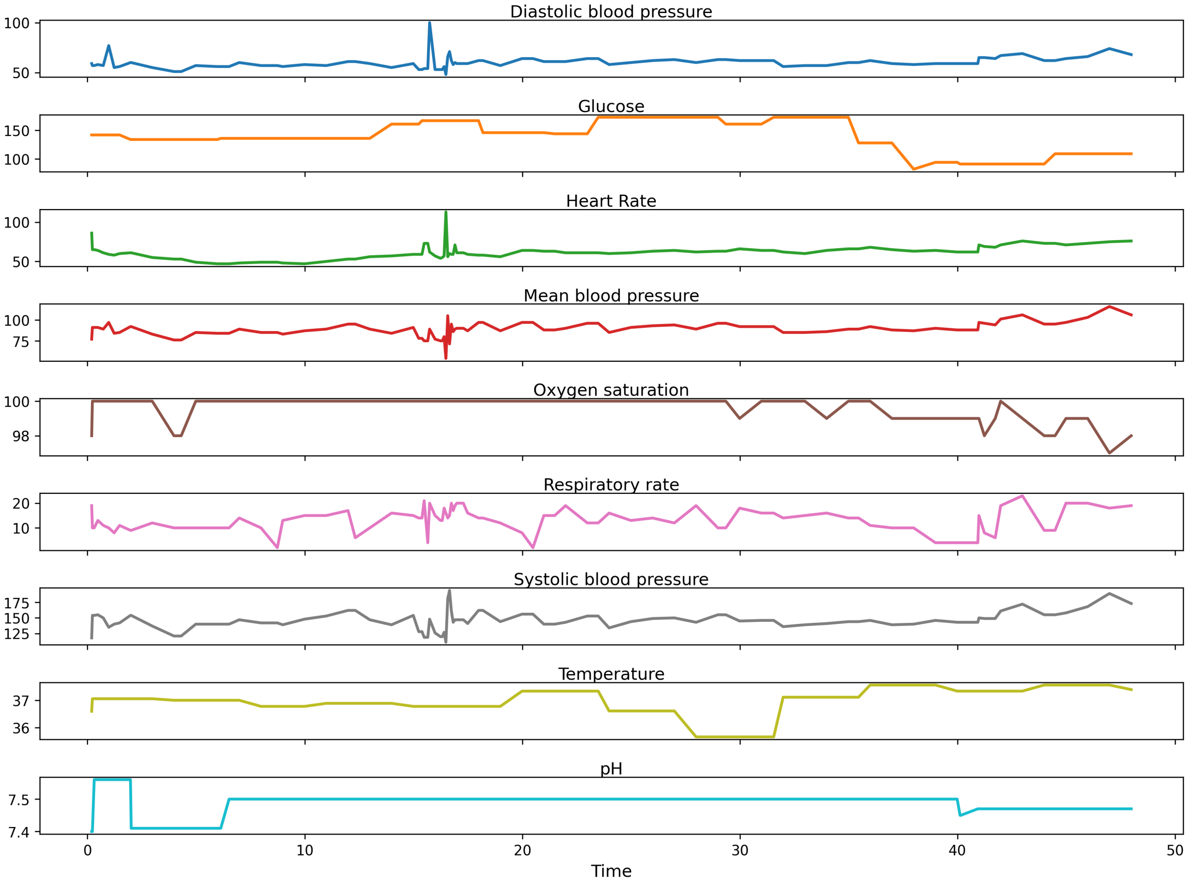
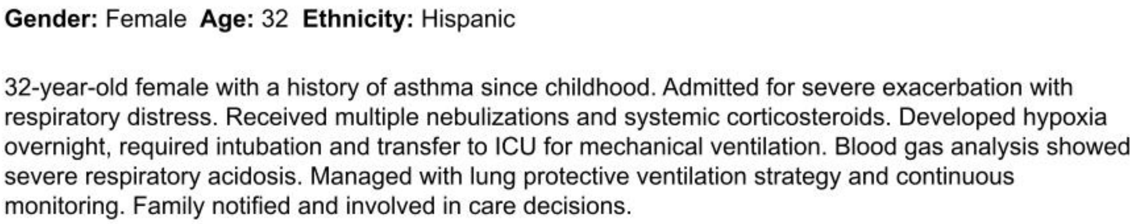


Figure 3: A synthetic sample of a patient's time-series in the MIMIC-III dataset.



Gender: Female **Age:** 32 **Ethnicity:** Hispanic

32-year-old female with a history of asthma since childhood. Admitted for severe exacerbation with respiratory distress. Received multiple nebulizations and systemic corticosteroids. Developed hypoxia overnight, required intubation and transfer to ICU for mechanical ventilation. Blood gas analysis showed severe respiratory acidosis. Managed with lung protective ventilation strategy and continuous monitoring. Family notified and involved in care decisions.

Figure 4: A synthetic sample of a patient's clinical text in the MIMIC-III dataset.

Optimizing Multimodal Large Language Models for Detection of Alcohol Advertisements via Adaptive Prompting

Daniel Cabrera Lozoya, Jiahe Liu
Simon D'Alfonso, and Mike Conway
The University of Melbourne, Australia
{dcabreralozo, jiahe3}@student.unimelb.edu.au
{dalfonso, mike.conway}@unimelb.edu.au

Abstract

Adolescents exposed to advertisements promoting addictive substances exhibit a higher likelihood of subsequent substance use. The predominant source for youth exposure to such advertisements is through online content accessed via smartphones. Detecting these advertisements is crucial for establishing and maintaining a safer online environment for young people. In our study, we utilized Multimodal Large Language Models (MLLMs) to identify addictive substance advertisements in digital media. The performance of MLLMs depends on the quality of the prompt used to instruct the model. To optimize our prompts, an adaptive prompt engineering approach was implemented, leveraging a genetic algorithm to refine and enhance the prompts. To evaluate the model's performance, we augmented the RICO dataset, consisting of Android user interface screenshots, by superimposing alcohol ads onto them. Our results indicate that the MLLM can detect advertisements promoting alcohol with a 0.94 accuracy and a 0.94 F1 score.

1 Introduction

The exposure of adolescents to advertisements promoting addictive substances is a risk factor for the subsequent development of maladaptive substance use patterns (Jackson et al., 2018). In the case of alcohol, exposure to alcohol advertising and the level of endorsement for alcohol-related advertisements among twelve-year-olds significantly affect the severity of alcohol-related issues experienced by individuals at age fifteen (Grenard et al., 2013). This impact is mediated by the escalation in alcohol consumption during this age period. Historically, studies examining the connection between exposure to addictive substance marketing and early use initiation among teenagers has predominantly centred on well-established mediums like television and newspapers (Anderson et al., 2009). However, the marketing landscape has evolved, with social media and web platforms emerging as dominant sources for advertising addictive substances. This shift is attributed to the under-regulation of these platforms and their widespread popularity among teenagers (Jackler et al., 2018; Zewude et al., 2022; Clendennen et al., 2020). In addition to advertisements sponsored by alcohol companies, there is a proliferation of user-generated content actively promoting the consumption of these substances. This phenomenon results in socially amplified advertising on social networking sites, presenting challenges in terms of regulation and monitoring (Salimian et al., 2014; Barry et al., 2018).

Multimodal Large Language Models (MLLMs) can process data from multiple modalities, such as text, images, and audio. In this study, we employed an MLLM to automate the detection of alcohol advertisements within digital media. Similar to Large Language Models (LLMs), the efficacy of an MLLM is contingent upon the instructive prompt's quality (Grabb, 2023). While substantial efforts have been directed toward prompt engineering for models that can only process text (Wei et al., 2022; Chen et al., 2023; Zelikman et al., 2022; Fernando et al., 2023), the exploration of prompt engineering for models capable of handling both text and images remains relatively underexplored. To optimize the instruction prompt for our MLLM, we employed a genetic algorithm for prompt generation and selection. Each of the instruction prompts represented an individual in our genetic algorithm. Through an iterative process of mutating and reproducing the fittest prompts, we identified the one yielding the best results. Each of the instruction prompts were crafted based on the following prompt engineering techniques: Chain-of-Thought (CoT) (Wei et al., 2022), Generated Knowledge (GK) prompting (Liu et al., 2022), Self-critique (Wang et al., 2023), and Expert prompting (Xu et al., 2023). Thus, our research also provides insights into the effectiveness of different prompt

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 514–525
August 16, 2024. ©2024 Association for Computational Linguistics

engineering techniques for MLLMs.

To evaluate the performance of our model, we augmented the RICO dataset (Deka et al., 2017). The RICO dataset comprises screenshots of user interfaces from various Android apps, such as social, dating, and communication apps. To augment the dataset, we incorporated advertisements from alcohol companies by superimposing them onto the RICO images. The MLLM was employed to classify the images based on the presence or absence of alcohol ads. The evaluation involved measuring the accuracy and F1 score of the classifier.

Our main contributions are as follows:

1. Development of a dataset of user interface screenshots with alcohol ads.

2. Creation, evaluation, and release of our adaptive prompt engineering algorithm for multimodal models. Our evaluation provides insights regarding which prompt engineering technique works best for MLLMs.

2 Related Work

2.1 Detection of addictive substances in digital media

The proliferation of alcohol advertisements on social media platforms has played a significant role in fostering maladaptive drinking behaviors among adolescents (Berey et al., 2017). As a result, multiple studies have aimed to develop effective tools for systematically monitoring the portrayal of alcoholic beverages and other addictive substance use within social media content. For example, Shanmugam et al. (2022) utilized the Darknet Framework and YOLOv3 for developing a parental control mobile application. This app enhanced monitoring of children's exposure to inappropriate content including substance use-related content on mobile devices, achieving an accuracy of 0.87 and an average precision score of 0.84. Hashmi et al. (2021) used a Mask R-CNN, Cascade Mask-R-CNN, and Hybrid Task Cascade to detect smoking images. Their best performing model, Mask R-CNN, achieved an average precision of 0.79 at an Intersection over Union (IoU) of 0.5. Using a further approach, Yang and Luo (2017) utilized a multimodal analysis method, employing multitask learning and decision-level fusion to identify drug-related posts on Instagram. Their best performing model achieved a precision of 0.83 in the task of recognizing drug-related posts. Pramanick et al. (2021) introduced the MOMENTA framework, a novel deep neural network approach that integrates VGG-19, CLIP Image Encoder, CLIP Text Encoder, and DistilBERT with self-attention mechanisms, for detecting alcohol-related harmful content in memes, achieving an accuracy of 0.83 and F1 score of 0.83. Ha et al. (2023) created a dataset focused on detecting harmful objects across six categories: alcohol, blood, cigarettes, guns, insulting gestures, and knives. This study showcased the enhanced detection capabilities of YOLOv5 and Faster R-CNN models, as evidenced by YOLOv5 achieving the highest mean average precision (mAP) of 0.94, while Faster R-CNN achieved a maximum mAP of 0.81 across all categories.

In contrast to previous approaches, our model is capable of identifying harmful content, even if presented in textual form. Additionally, unlike earlier models that evaluated independent images to determine if the entire image was associated with harmful content (Hashmi et al., 2021; Yang and Luo, 2017; Shanmugam et al., 2022; Pramanick et al., 2021; Ha et al., 2023), our approach also discerns harmful elements within discrete portions of an image. This distinction holds particular importance, given that advertisements featuring harmful content may not always dominate the entire screen; they could be confined to small sections within the overall image. The ability to detect harmful content in discrete portions of an image provides flexibility compared to other models. Unlike previous methods that relied on first extracting all web image elements from a site and then using classifiers to identify harmful content (Chou et al., 2008; Invernizzi et al., 2016), our approach is more adaptable. This adaptability is particularly valuable in the context of live stream videos, a format that has gained popularity in social media (Zimmer, 2018). In contrast to preloaded and static content, such as images, live stream videos pose a significant challenge to substance use image detection systems due to their real-time and dynamic nature.

2.2 Prompt Engineering

The effectiveness of language models in completing tasks depends on the quality of the prompts they receive (Grabb, 2023). Strategies in prompt engineering, such as CoT, Graph of Thoughts (Besta et al., 2023) and thought decomposition (Xie et al., 2023), involve incorporating intermediate steps to enhance a model's problem-solving capabilities. Promoting a diverse set of intermediate steps is a critical aspect when optimizing prompts, since it enables a model to explore a vast solution space for effective problem-solving (Fernando et al., 2023). Highlighting the impact of prompt diversity on model performance, self-consistency (Wang et al., 2022) boosted the performance of CoT by replacing the naive greedy strategy employed in CoT. In self-consistency, a diverse set of intermediate steps are initially sampled, as opposed to always opting for the immediately best one. Subsequently, the model selects the most consistent answer from this varied set of intermediate steps. By leveraging the intuition that a complex reasoning problem admits a diverse set of intermediate steps, self consistency boost the performance of CoT on a range of popular arithmetic benchmarks, such as GSM8K (+17.9%) and SVAMP (+11.0%). Similarly, Auto-Cot (Zhang et al., 2022) underscores the importance of diversity in intermediate reasoning steps to enhance LLMs. By diversifying these steps, Auto-Cot consistently matched the performance of manually crafted CoT across ten public benchmarks.

Automated prompt strategies, aimed at minimizing manual intervention in prompt design and optimization, have demonstrated promising results. For instance APE, an Automated Prompt Engineering (Zhou et al., 2022) scheme, achieved human-level performance on the 17/21 Big-Bench and the Instruction Induction datasets. APE leverages LLMs to generate task-prompts candidates and to introduce prompt mutation to add variability to the task-prompts employed for problem-solving. In our study, we adopted the methodology employed in PromptBreeder (Fernando et al., 2023), which aims to enhance diversity within prompts by modifying both the prompts responsible for mutating instruction prompts and the instruction prompts themselves. The Promptbreeder approach uses a binary tournament genetic algorithm framework (Harvey, 2009). This entails randomly selecting two prompts originating from different instruction tasks, and replacing the prompt with the lower fitness by a mutated version of the one with the higher fitness.

Given that PromptBreeder consistently opts for the prompt with the highest fitness at each stage, this greedy approach introduces the risk of getting trapped in a local maximum. Greedy algorithms tend to converge faster than their non-greedy counterpart, this characteristic poses a challenge in the realm of automated prompt engineering. The rapid convergence results in prompts resembling only those with the highest fitness, thus reducing the diversity of prompts and limiting the search exploration for the optimal one. To prevent convergence to a local maximum, a distinct heuristic was employed for winner selection in the genetic tournament. We used the roulette wheel selection method to select the individuals for the next generations (Behera, 2020). Instead of solely relying on individual fitness, we normalized the overall fitness of all prompts. The normalized value is then used in a probability function to select the winner. This method maintains a preference for prompts with higher fitness, while granting prompts with lower fitness an opportunity to mutate and potentially contribute to the solution by exploring alternative paths that might lead to the optimal outcome. This method promotes a more balanced exploration of the solution space by increasing the diversity of the prompts.

Previous prompt engineering techniques were predominantly either manually crafted or exclusively evaluated for Large Language Models. In this research, we are pioneering an automated prompt engineering technique tailored for a Multimodal Large Language Model. Notably, the mutation prompts utilized to evolve the task prompts are rooted in successful prompt engineering techniques previously designed for LLMs. We systematically track the performance of these mutation prompts, providing valuable insights into their efficacy within the context of MLLMs. This approach allows us to discern and adapt what proves to be effective for enhancing the performance of MLLMs.

3 Method

3.1 Data collection

To construct our training and testing dataset, we utilized the RICO dataset (Deka et al., 2017) by extracting 2,100 distinct user interface (UI) screenshots from it. Additionally, we employed a web scraper to gather images from Google featuring alcohol advertisements. Please see Appendix A for the terms used to search for alcohol ad images. An author of the paper reviewed the downloaded images to remove non-alcohol-related ones, resulting in a curated dataset of 2,100 different alcohol ad images. These advertisement images were resized

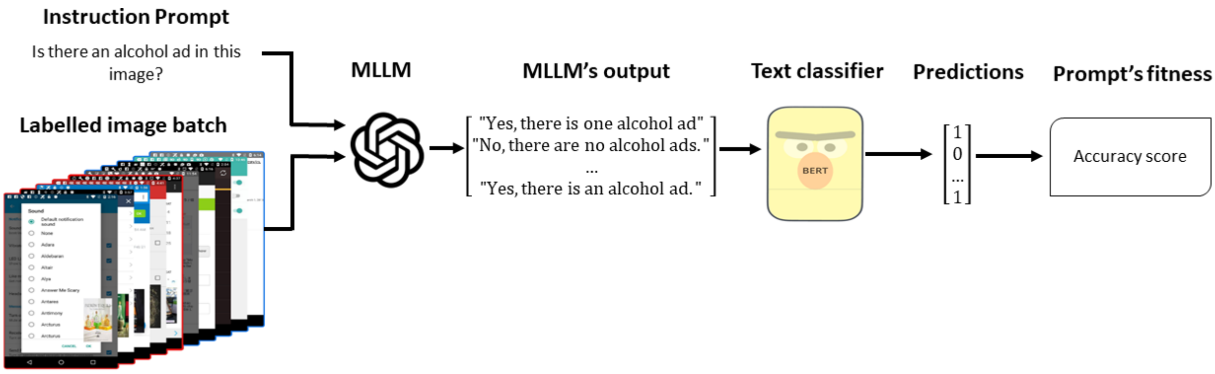

Instruction Prompt
Is there an alcohol ad in this image?
Labelled image batch
MLLM
MLLM's output
"Yes, there is one alcohol ad"
"No, there are no alcohol ads."
...
"Yes, there is an alcohol ad."
Text classifier
BERT
Predictions
1
0
...
1
Prompt's fitness
Accuracy score

Figure 1: To evaluate a prompt's fitness, we use an MLLM with the prompt and a batch of labeled images as input. The output from the MLLM is then labeled by a binary text classifier. The resulting accuracy represents the prompt's fitness.

to one-eighth of the UI images and superimposed onto them. Please refer to Figure 5 in Appendix B for an example of a superimposed image. Consequently, the resultant dataset comprised 4,200 images, which were partitioned into a stratified training and testing datasets, allocating 3,200 for training and 1,000 for testing. The training dataset was then divided into batches of 200 image each, each one containing an equal number of images with and without alcohol ads.

3.2 Genetic Algorithm

Let O represent the output from an MLLM when given an instruction prompt T and an image I as inputs, expressed as $O = \mathrm{MLLM}(T, I)$. Our genetic algorithm aims to find an optimal instruction prompt P with the goal of maximizing the quality of O in comparison when T is utilized.

Similar to PromptBreeder, our algorithm mutates prompts to optimize them. Mutations involve a mutation prompt M and an LLM. A mutated prompt P' is defined as P' = LLM(M+ P), where + denotes string concatenation. The pool of mutation-prompts is elaborated upon in section 3.4. Refer to Appendix C for a prompt mutation example. Mutation-prompts are also evolved via hypermutations (Ouertani et al., 2019). To do so a hypermutation prompt H and an LLM are used. An evolved mutation-prompt M' is represented as M' = LLM(H + M).

Given an initial instruction prompt consisting of detecting alcohol ads in an image, our algorithm creates an initial population of prompts by evolving the initial instruction prompt using a set of random mutation prompts. The mutated prompts are then used by the MLLM to make predictions on a random batch from the training dataset. Once the batch has been processed, the detection accuracy that the MLLM got using each prompt is stored as the fitness level of that prompt. Our algorithm maintains a record of the instruction prompt, the mutation prompt, and the associated fitness level that the prompt achieved when processing a batch of images. Each record represents an individual in the population.

Once the population is initialized, our evolutionary process unfolds in generational iterations. In each generation, each individual has a mutation probability of μ_m, representing the likelihood of undergoing a mutation that alters its instruction prompt. After selecting which individuals will undergo a mutation, our algorithm then determines the type of mutation to be acquired from four options: Chain of Thought, Generated Knowledge, Self-verification, or Expert Prompting. To strike a balance between breadth and depth needed for a robust evolutionary search (Moreno-Bote et al., 2020), each mutation mechanism initially has an equal base probability of being the acquired mutation. However, as generations progress, mutation types with a proven track record of producing superior fitness outcomes are granted an increased chance in addition to the base probability.

Upon calculating the mutated individual's fitness using a random batch from the training dataset, it is introduced into the population. This iterative process continues until the maximum population cap is reached. Upon reaching the population cap, succeeding generations employ a roulette wheel selection method to determine individuals advancing to the subsequent generation and those being phased out. To mitigate the risk of falling into a local maxima, our algorithm samples the surviv-

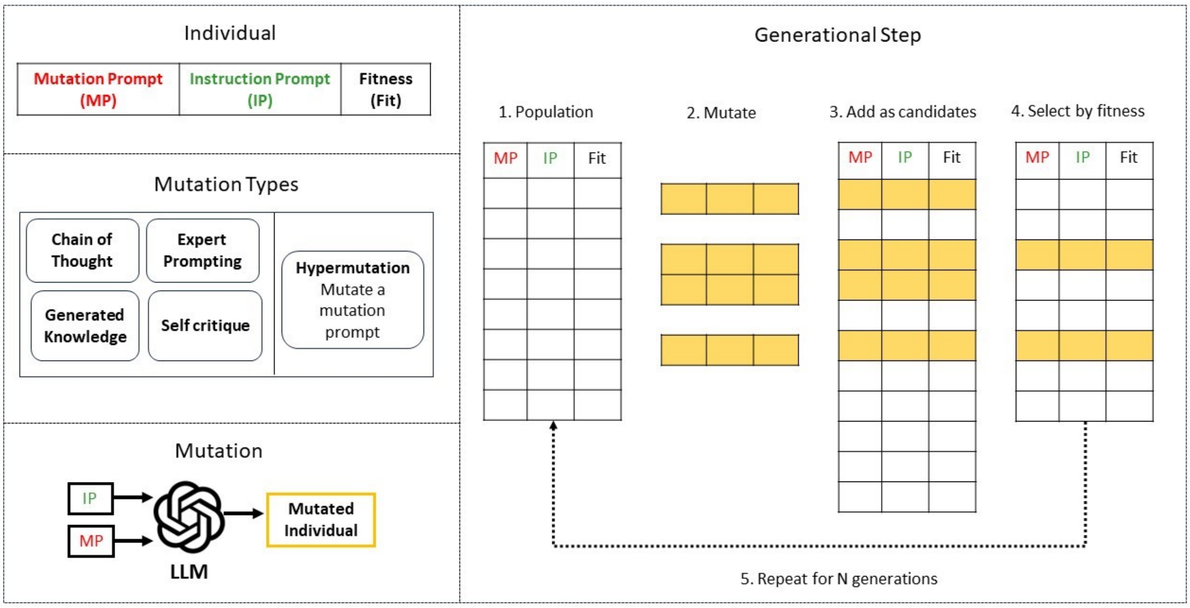


Figure 2: In our genetic algorithm, individuals consist of three components: an instruction prompt for guiding the MLLM, a mutation prompt that was used to generate the instruction prompt, and a corresponding fitness determined by the accuracy of the MLLM's performance using that prompt. At the beginning of the algorithm, the initial population is formed, with one individual generated for each mutation type. During each generational step, there is a probability for each individual to undergo a mutation that modifies its instruction prompt. The specific mutation type is chosen from a mutation pool. The individuals that experience mutation are then incorporated into the population. When the maximum population cap is reached, a fitness-based probabilistic selection is employed to determine which individuals progress to the next generation.

ing individuals using a probability based on their fitness (Marsili Libelli and Alba, 2000). While fitter individuals possess a higher likelihood of survival, underperforming individuals, with potential for uncovering global maxima, are still given an opportunity to contribute to forthcoming generations. After N generations, the instruction prompt from the individual with the highest fitness is selected as the optimized prompt. Figure 2 presents an overview of our algorithm.

3.3 Natural Language Processing Models

Our genetic algorithm was tested using two types of models, one open-source and one proprietary. The open-source MLLM we used was LLaVA (Large Language and Vision Assistant), its code being licensed under the Apache License 2.0. The selection of the LLaVA model was driven by its capability to be run locally. This attribute is particularly crucial for applications of this nature, where the analysis involves social media images that may contain sensitive and personal information from users. The ability to execute the model locally enhances privacy and security considerations in handling such data. For the proprietary MLLM, we utilized OpenAI's model 'gpt-4-vision-preview'. The choice of OpenAI models was motivated by their superior performance compared to open source alternatives. The MLLMs received as input an image and an instruction prompt instructing them to identify any advertisements for alcohol within the image.

Since the MLLMs can generate diverse textual outputs to indicate the presence or absence of such ads, we appended a formatting prompt to the instruction prompt, requesting the model to respond with a 'yes' or 'no'. Subsequently, a BERT text classifier was utilized to categorize the MLLM's outputs. A label of 0 was assigned to responses indicating no alcohol ad content, while a label of 1 was assigned to responses indicating the presence of alcohol ads, as demonstrated in Figure 1. This classification step ensures a standardized and consistent output, which was needed to measure the performance of the MLLM model. To train the BERT classifier the MLLM processed one image batch from our training dataset. Subsequently, we leveraged OpenAI's GPT-3.5 Turbo model for data augmentation, generating a total of 10,000 texts, with half affirming the presence of harmful content and the other half negating it. We then divided the augmented outputs into a training and testing dataset, with a distribution of 80% to 20% respectively. We used an Adam optimizer with weight decay, using a learning rate of 1.0×10^{-5}, and trained it for 10 epochs. The accuracy of the BERT classifier was 0.98.

For prompt optimization, we used OpenAI's GPT-3.5 Turbo model. The computing infrastructure employed for running all the NLP models was an NVIDIA A100 GPU.

3.4 Mechanisms of mutation

The pool of mutation prompt types is derived from prompt engineering techniques employed to enhance prompts for LLMs. Refer to Appendix D for the set of starting prompts for each type of mutation.

3.4.1 Chain-of-Thought

Chain-of-Thought (CoT) is a prompt engineering technique that leverages task decomposition to enhance a model's performance. This approach involves introducing intermediate reasoning steps, enabling LLMs to undertake intricate reasoning tasks. In our implementation, we utilized the zero-shot version of Chain-of-Thought, as described by Kojima et al. (Kojima et al., 2023). Specifically, this technique appends variations of the string "Let's think step by step" to the original prompt.

3.4.2 Generated knowledge prompting

Generated Knowledge prompting involves a two-phase process designed to enhance the performance of an LLM. The first phase is the knowledge generation stage, where a language model is tasked with producing additional valuable information pertinent to a specific task. Subsequently, in the knowledge integration phase, a second language model utilizes this additional information as input to carry out its designated task.

3.4.3 Self-critique

Self-critique is a two-step process designed to improve the output of an LLM by inspecting and criticizing its own initial output. The initial stage involves forward reasoning, where the model utilizes a prompt to address a specific task. In the subsequent backward-verification phase, a second LLM scrutinizes the validity of the initial answer.

3.4.4 Expert prompting

Expert prompting involves explicitly indicating to an LLM that it is proficient in a particular field. In our scenario, where the goal is to create a versatile genetic algorithm applicable to various tasks, we inform the model that its expertise lies in prompt engineering tasks. Figure 6 in Appendix C illustrates an example of a mutation using an expert prompting technique.

3.4.5 Hyper-mutation

A hyper-mutation occurs when a mutation prompt is mutated, thereby expanding the dimensions of the search space for each prompt. To execute this process, we select a mutation prompt from one of our mutation prompt pools and utilize it to modify another prompt from the same pool. For instance, we can mutate a CoT mutation prompt by employing another CoT mutation prompt. Subsequently, this newly generated mutation prompt is incorporated into its corresponding mutation prompt pool.

3.5 Evaluation

To determine the optimal task prompt, we executed the genetic algorithm with a population limit set to 20 individuals, a mutation probability μ_m of 50%, and spanning a total of 15 generations. Subsequently, we selected the prompt with the highest fitness level from the surviving population. The selected prompt became the input for the MLLM, and we assessed its performance using the images from the testing dataset. Our evaluation metrics included measuring and reporting both the F1-score and the accuracy achieved by the MLLM on the testing dataset.

4 Results

In this section, we present our findings derived from the evaluation tasks. The subsequent section is dedicated to a comprehensive discussion and analysis of the implications arising from these outcomes. The optimized prompts ultimately resulting from the genetic algorithms were as follows:

1. **GPT-4:** *Examine the image carefully for visual cues of alcohol, such as beer bottles, wine glasses, or alcohol brand logos. Pay attention to color, shape, and context that may indicate alcohol consumption. Consider the setting of the image - is it a bar, party, or other alcohol-present environment? Look for behavioral cues like people holding drinks or appearing*

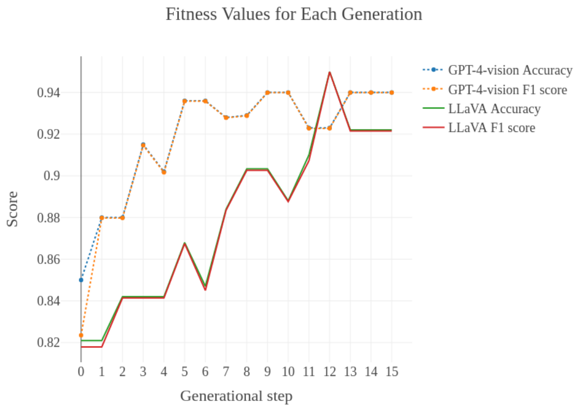


Figure 3: Performance metrics: Accuracy and F1 scores for GPT-4 and LLaVA in alcohol advertisement detection.

intoxicated, and note any subtle elements suggesting alcohol reference. Answer with a yes or a no.

2. **LLaVA:** *Does the following image contain an alcohol ad? Carefully analyze the image for any alcohol brands, such as beer logos, wine labels, or liquor bottles. Pay attention to distinctive colors or shapes commonly found on alcohol packaging. Make note of labels, bottles, or glasses that suggest the presence of alcoholic beverages. Approach the task systematically, considering each element one by one. Answer with a yes or a no.*

Figure 3 depicts the performance metrics of the MLLMs employing an optimized prompt generated through our genetic algorithm. After 15 generational steps, the classifier utilizing the GPT-4 vision model as the MLLM obtained an accuracy of 0.94 and an F1-score of 0.94. The classifier employing the LLaVA model as the MLLM achieved an accuracy of 0.922 and an F1-score of 0.9215.

In Figure 4, the distribution of mutation types among individuals across generations is illustrated for the genetic algorithm employing GPT-4 and LLaVA. For the GPT-4 model, the CoT mutation type consistently generated prompts that were selected to advance to the next generation through the roulette wheel selection method. In the case of the genetic algorithm utilizing the LLaVA model, CoT and Generated Knowledge were the mutation types with the highest-frequency of occurrence that persisted in each generation.

5 Discussion

The most effective prompts and prevalent mutation types observed throughout multiple generations stemmed from the CoT prompt engineering technique, with the top-performing prompts from the final generation being a product of a CoT mutation prompt. However, upon examining the prompts, we noted their integration of elements from different prompt engineering methods. Prompts created from the generated knowledge mutation prompts consistently include enumerations of components for image inspection, as shown in this optimized prompts. Therefore, our findings suggest that the optimal prompt engineering approach involves a blend of different techniques.

The performance of the open-source model in detecting alcohol ads in images is comparable to that of the proprietary model. This is a promising result, as it enables researchers to analyze sensitive images without the necessity of sending them to third-party organizations. Moreover, the fact that the model is open-source potentially reduces costs, hence increasing accessibility to the tools in less well-resourced settings.

Our adaptive prompt engineering technique presents a more accessible approach for public health researchers seeking to apply automated methods to the identification of other types of harmful online content. Notably, our method reduces the need for users to possess a background in machine learning for training to optimize an MLLM. Additionally, it operates without reliance on the model's proprietary weights or architecture, which can be inaccessible. Furthermore, users are not required to possess prompt engineering experience, as state-of-the-art prompt engineering techniques are already integrated into the algorithm. Moreover, our algorithm allows for easy upgrades upon the discovery of new prompt engineering techniques, requiring only their addition to the mutation pool.

6 Conclusion

We developed a genetic algorithm to optimize the prompt for MLLMs to detect harmful content in images. We also extended the RICO dataset which contains UI screenshots by superimposing alcohol advertisements. The optimal prompt achieved an accuracy score of 0.94 and a F1 score of 0.94.

The mutation prompts utilized in our algorithm were derived from prompt engineering techniques traditionally employed for LLMs. However, these

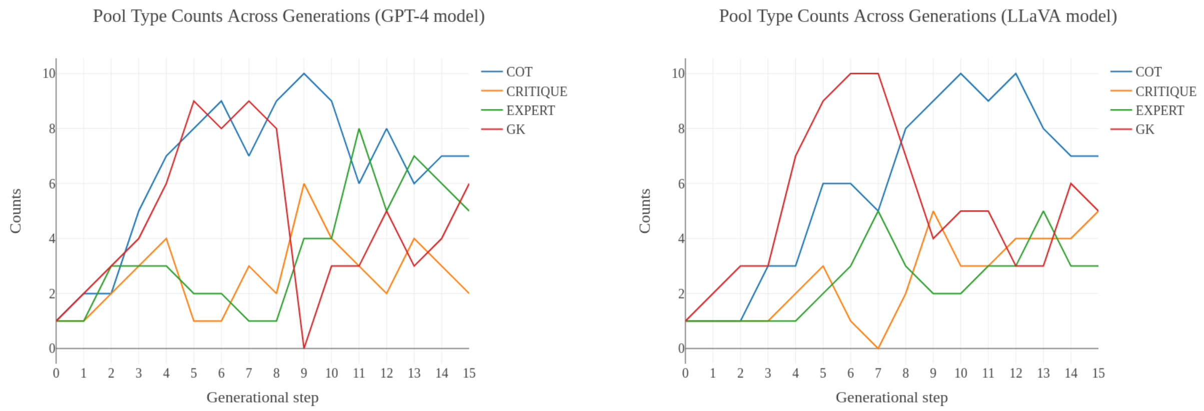


Figure 4: Number of individuals from a given mutation type in the population at a given generational step. Left: GPT-4 model; Right: LLaVA model.

approaches had not been previously tested within the framework of MLLMs. By tracking the performance of each mutation type, we identified that, within the realm of MLLMs, the CoT and the Generated Knowledge mutations outperformed the Expert and self-critique approaches.

Although our algorithm was initially designed and tested for detecting alcohol advertisements, it can be extended to identify other harmful substances such as tobacco and drugs when provided with the appropriate data sets. We envision that by adapting our algorithm online platforms can detect and remove harmful content, thereby fostering a safer online environment.

7 Limitations

Our main objective in implementing the genetic algorithm was to identify prompts that optimized the MLLM for detecting harmful content in images. However, the optimization strategy does not explicitly address potential biases introduced by the chosen prompts. For example, if the training examples lead the model to establish an inaccurate association between an ethnic group and alcohol consumption, it could result in the creation of biased prompts. Generative models may exhibit biases in their outputs, requiring a comprehensive examination to mitigate the inadvertent propagation of such biases (Hemmatian and Varshney, 2022; Abid et al., 2021; Cabrera Lozoya et al., 2023).

Due to resource constraints associated with using a paid MLLM, we faced limitations in conducting additional experiments to evaluate the robustness of our models. Various hyperparameters could have been explored, such as adjusting the mutation rate, maximum population size, or the number of generations employed to discover the optimal prompt. Additionally, both GPT-3.5 Turbo and GPT-4 Vision possess the capability to handle multiple languages. However, our collection of ads exclusively consisted of English ads. Furthermore, due to hardware constraints, we opted for the 7 billion LLaVA model, despite the existence of larger models that outperform the one chosen. Consequently, this decision limits our ability to demonstrate the potential of an open-source model for detecting harmful content.

While the detection of alcohol advertisements serves to protect vulnerable populations, notably teenagers, from the impact of marketing materials on their attitudes and behaviors related to alcohol consumption, the utilization of such technologies carries inherent risks of improper use. There is a potential for entities to exploit the technology beyond its intended public health purpose, conducting surveillance or accessing sensitive information, thus posing a threat to privacy and civil liberties. Hence, the application of our image detector requires a balanced ethical framework. Achieving a careful balance is crucial, seeking to maximize the tool's positive contributions to public health while actively addressing potential concerns through robust privacy safeguards, bias mitigation, and responsible deployment practices.

References

Abubakar Abid, Maheen Farooqi, and James Zou. 2021. Persistent anti-muslim bias in large language models. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '21, page 298–306, New York, NY, USA. Association for Computing Machinery.

P. Anderson, A. de Bruijn, K. Angus, R. Gordon, and G. Hastings. 2009. Impact of alcohol advertising and media exposure on adolescent alcohol use: A systematic review of longitudinal studies. *Alcohol and Alcoholism*, 44(3):229–243.

Adam E. Barry, Alisa A. Padon, Shawn D. Whiteman, Kristen K. Hicks, Amie K. Carreon, Jarrett R. Crowell, Kristen L. Willingham, and Ashley L. Merianos. 2018. Alcohol advertising on social media: Examining the content of popular alcohol brands on instagram. *Substance Use amp; Misuse*, 53(14):2413–2420.

Narayan Behera. 2020. *Analysis of microarray gene expression data using information theory and stochastic algorithm*, page 349–378. Elsevier.

Benjamin L Berey, Cassidy Loparco, Robert F Leeman, and Joel W Grube. 2017. The myriad influences of alcohol advertising on adolescent drinking. *Current addiction reports*, 4:172–183.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Michał Podstawski, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. 2023. Graph of Thoughts: Solving Elaborate Problems with Large Language Models.

Daniel Cabrera Lozoya, Simon D'Alfonso, and Mike Conway. 2023. Identifying gender bias in generative models for mental health synthetic data. In *2023 IEEE 11th International Conference on Healthcare Informatics (ICHI)*, pages 619–626.

Siyuan Chen, Mengyue Wu, Kenny Q. Zhu, Kunyao Lan, Zhiling Zhang, and Lyuchun Cui. 2023. Llm-empowered chatbots for psychiatrist and patient simulation: Application and evaluation.

Yao-Ping Chou, Shi-Jinn Horng, Hung-Yan Gu, Cheng-Ling Lee, Yuan-Hsin Chen, and Yi Pan. 2008. Detecting pop-up advertisement browser windows using support vector machines. *Journal of the Chinese Institute of Engineers*, 31(7):1189–1198.

Stephanie L. Clendennen, Alexandra Loukas, Elizabeth A. Vandewater, Cheryl L. Perry, and Anna V. Wilkinson. 2020. Exposure and engagement with tobacco-related social media and associations with subsequent tobacco use among young adults: A longitudinal analysis. *Drug and Alcohol Dependence*, 213:108072.

Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. 2017. Rico: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, page 845–854, New York, NY, USA. Association for Computing Machinery.

Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. 2023. Promptbreeder: Self-referential self-improvement via prompt evolution.

Declan Grabb. 2023. The impact of prompt engineering in large language model performance: a psychiatric example. *Journal of Medical Artificial Intelligence*, 6(0).

Jerry L. Grenard, Clyde W. Dent, and Alan W. Stacy. 2013. Exposure to alcohol advertisements and teenage alcohol-related problems. *Pediatrics*, 131(2):e369–e379.

Eungyeom Ha, Heemook Kim, Sung Chul Hong, and Dongbin Na. 2023. HOD: A Benchmark Dataset for Harmful Object Detection.

Inman Harvey. 2009. The microbial genetic algorithm. In *European Conference on Artificial Life*.

Muhammad Umer Hashmi, Ngoc Duy Nguyen, Michael Johnstone, Kathryn Backholer, and Asim Bhatti. 2021. *Application Based Cigarette Detection on Social Media Platforms Using Machine Learning Algorithms*, page 68–80. Springer International Publishing.

Babak Hemmatian and Lav R. Varshney. 2022. Debiased large language models still associate muslims with uniquely violent acts.

Luca Invernizzi, Kurt Thomas, Alexandros Kapravelos, Oxana Comanescu, Jean-Michel Picod, and Elie Bursztein. 2016. Cloak of Visibility: Detecting When Machines Browse a Different Web. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 743–758, San Jose, CA. IEEE.

Robert K Jackler, Vanessa Y Li, Ryan A L Cardiff, and Divya Ramamurthi. 2018. Promotion of tobacco products on facebook: policy versus practice. *Tobacco Control*, pages tobaccocontrol–2017–054175.

Kristina M. Jackson, Tim Janssen, and Joy Gabrielli. 2018. Media/marketing influences on adolescent and young adult substance abuse. *Current Addiction Reports*, 5(2):146–157.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2023. Large language models are zero-shot reasoners.

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. 2022. Generated knowledge prompting for commonsense reasoning. In *Proceedings of the*

60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 3154–3169, Dublin, Ireland. Association for Computational Linguistics.

Stefano Marsili Libelli and P Alba. 2000. Adaptive mutation in genetic algorithms. *Soft computing*, 4:76–80.

Rubén Moreno-Bote, Jorge Ramírez-Ruiz, Jan Drugowitsch, and Benjamin Y Hayden. 2020. Heuristics and optimal solutions to the breadth–depth dilemma. *Proceedings of the National Academy of Sciences*, 117(33):19799–19808.

Nasreddine Ouertani, Issam Nouaouri, Hajer Ben-Romdhane, Hamid Allaoui, and Saoussen Krichen. 2019. A hypermutation genetic algorithm for the dynamic home health-care routing problem. In *2019 International Conference on Industrial Engineering and Systems Management (IESM)*, pages 1–6. IEEE.

Shraman Pramanick, Shivam Sharma, Dimitar Dimitrov, Md Shad Akhtar, Preslav Nakov, and Tanmoy Chakraborty. 2021. MOMENTA: A Multimodal Framework for Detecting Harmful Memes and Their Targets.

Parissa K. Salimian, Rumi Chunara, and Elissa R. Weitzman. 2014. Averting the perfect storm: Addressing youth substance use risk from social media use. *Pediatric Annals*, 43(10).

Sathesh Ajith Kumar Hariharan Chandra Sekar Shanmugam, Maheswaran, Ridhish R, and Gomathi R D. 2022. YOLO based Efficient Vigorous Scene Detection And Blurring for Harmful Content Management to Avoid Children's Destruction. In *2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pages 1063–1073, Coimbatore, India. IEEE.

Rui Wang, Hongru Wang, Fei Mi, Yi Chen, Ruifeng Xu, and Kam-Fai Wong. 2023. Self-critique prompting with large language models for inductive instructions.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models.

Yuxi Xie, Kenji Kawaguchi, Yiran Zhao, Xu Zhao, Min-Yen Kan, Junxian He, and Qizhe Xie. 2023. Self-evaluation guided beam search for reasoning.

Benfeng Xu, An Yang, Junyang Lin, Quan Wang, Chang Zhou, Yongdong Zhang, and Zhendong Mao. 2023. Expertprompting: Instructing large language models to be distinguished experts.

Xitong Yang and Jiebo Luo. 2017. Tracking illicit drug dealing and abuse on instagram using multimodal analysis. *ACM Transactions on Intelligent Systems and Technology*, 8(4):1–15.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. 2022. Star: Bootstrapping reasoning with reasoning.

Bewunetu Zewude, Abebe Mengesha, and Sintayehu Temesgen. 2022. The impact of advertisements on adolescents' decision to consume beer: The case of selected high school students in shashemene town, west arsi zone. 3:81–86.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022. Automatic chain of thought prompting in large language models.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2022. Large language models are human-level prompt engineers.

Franziska Zimmer. 2018. *A Content Analysis of Social Live Streaming Services*, page 400–414. Springer International Publishing.

A Search terms

The following is the list of terms used to search for alcohol advertisements: Alcohol ads, Beer ads, Whiskey ads, Tequila ads, Lager ads, Ale ads, Red wine ads, White wine ads, Vodka ads, Stout ads, Scotch ads, Brandy ads, Champagne ads, Cider ads, Sake ads, Mezcal ads, Soju ads, Rosé ads, Rum ads, Gin ads, Cognac ads, Bailey irish cream ads, Grand Marnier ads, Amaretto ads, Khalúa ads, Triple Seca ads, Schnapps ads, Raki ads, Baijiu ads, Flavored Vodka ads, Extra añejo tequila ads, Blano tequila ads, Reposado tequila ads, Añejo tequila ads, Wheat vodka ads, Grappa ads, Pilsner ads, and Pisco ads.

B Image example

Figure 5 illustrates an example of an original UI screenshot from the RICO dataset, and a version with an alcohol ad superimposed.

C Mutation example

Figure 6 illustrates an example of a mutation step. In this scenario a mutated prompt is created by using a mutation prompt from the Expert pool to mutate an instruction prompt.

D Prompts

Table 1. presents the initial prompts for each type of mutation.

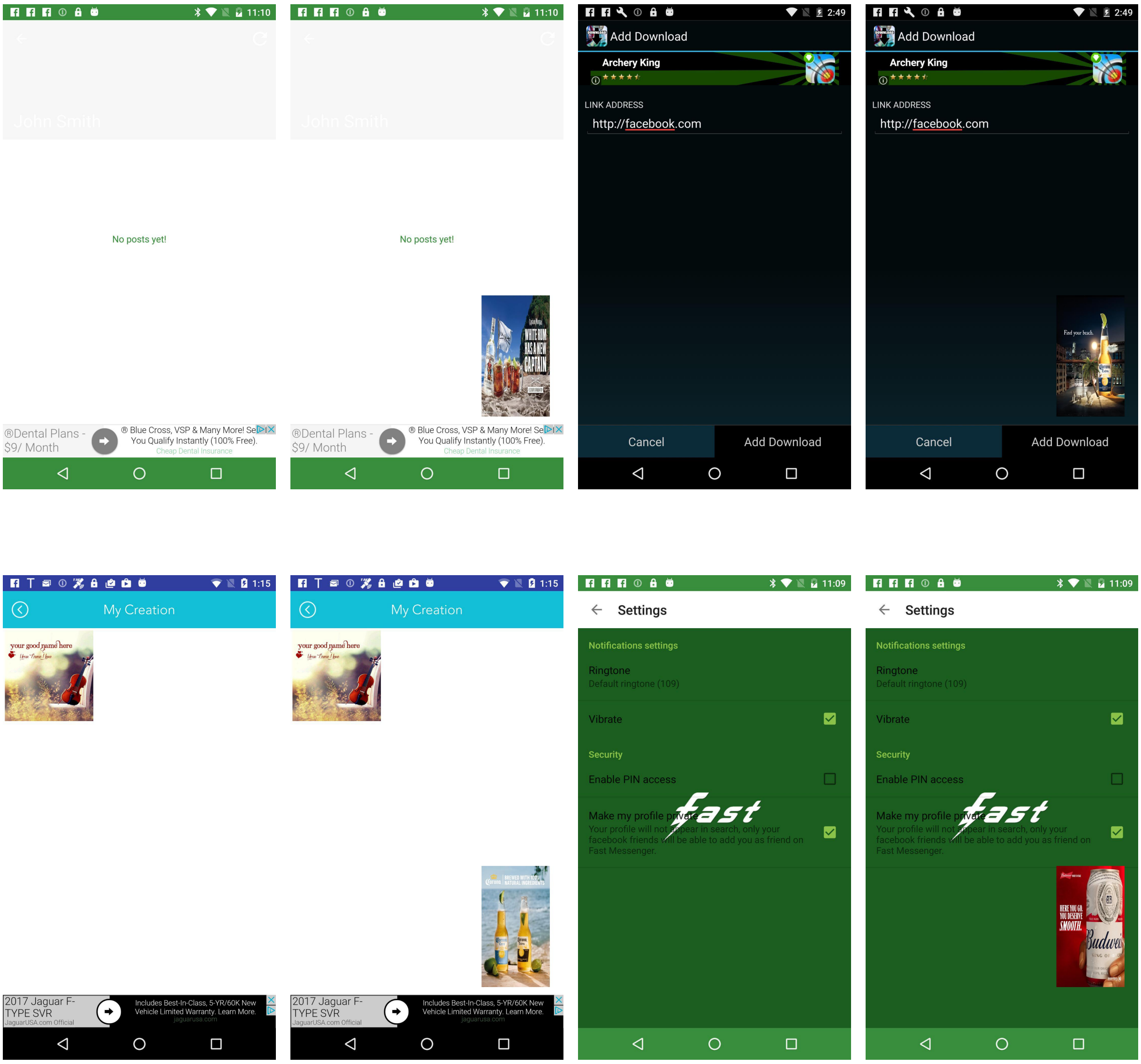
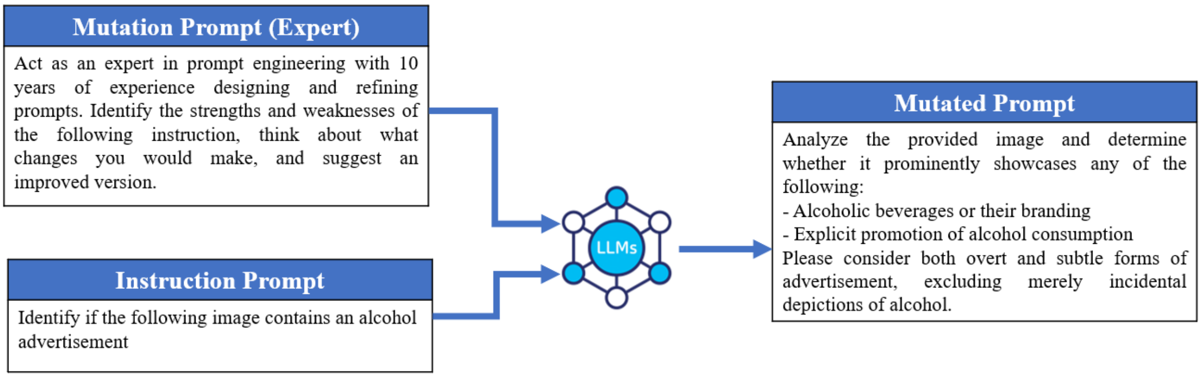


Figure 5: Examples of RICO UI screenshots and their modified version with an alcohol ad.



Mutation Prompt (Expert)

Act as an expert in prompt engineering with 10 years of experience designing and refining prompts. Identify the strengths and weaknesses of the following instruction, think about what changes you would make, and suggest an improved version.

Instruction Prompt

Identify if the following image contains an alcohol advertisement

Mutated Prompt

Analyze the provided image and determine whether it prominently showcases any of the following:
- Alcoholic beverages or their branding
- Explicit promotion of alcohol consumption
Please consider both overt and subtle forms of advertisement, excluding merely incidental depictions of alcohol.

Figure 6: Example of a mutation step.

Mutation type	**Prompts**
Chain of thought	Append to the following instruction the following text, "Let's think step by step."
	Decompose and rewrite the instruction as a set of logical steps, rewrite it as a sentence.
	Rewrite the following instruction by adding intermediate steps to enhance its performance.
Expert	Act as an expert in prompt engineering with 10 years of experience designing and debugging prompts. Identify the strengths and weaknesses of the following instruction, think about what changes you would make, and suggest an improved version.
	Imagine you are an expert in generating instructions for large multimodal models. You are designing an instruction to achieve the best possible result. A colleague shares their best instruction with you; identify why it is good and generate an even better one.
	Simulate being an expert program in improving instructions, detecting their strengths, weaknesses, and consistently providing better results. Take this prompt and make it better.
Generated Knowledge	Enhance the effectiveness of the following prompt by generating and appending additional content. Focus on providing specific examples, detailed criteria, or relevant guidelines to elevate its performance.
	Improve the prompt's performance through the strategic generation and integration of supplementary content, fostering heightened efficacy within the experimental domain.
	Optimize the prompt's performance via the meticulous generation and incorporation of additional content.
Critique	Critique the following instruction and propose enhancements to address any identified shortcomings. Please provide only the refined version in your response.
	Review the given instruction, identify any areas for improvement, and suggest changes to enhance its quality. Please provide a refined version that incorporate these improvements.
	Examine the given instruction, analyze it for potential shortcomings, and suggest improvements to address any identified issues. Submit only the refined version in your response, integrating enhancements to elevate its overall quality.

Table 1: Starting prompts for each mutation type.

Extracting Epilepsy Patient Data with Llama 2

Ben Holgate*, Shichao Fang*, Anthony Shek, Matthew McWilliam*, Pedro Viana*,**
Joel S. Winston*, James T. Teo*, and Mark P. Richardson*
* Department of Basic & Clinical Neuroscience, King's College London
** Guy's and St Thomas' NHS Foundation Trust
benjamin.holgate@kcl.ac.uk

Abstract

We fill a gap in scholarship by applying a generative Large Language Model (LLM) to extract information from clinical free text about the frequency of seizures experienced by people with epilepsy. Seizure frequency is difficult to determine across time from unstructured doctors' and nurses' reports of outpatients' visits that are stored in Electronic Health Records (EHRs) in the United Kingdom's National Health Service (NHS). We employ Meta's Llama 2 to mine the EHRs of people with epilepsy and determine, where possible, a person's seizure frequency at a given point in time. The results demonstrate that the new, powerful generative LLMs may improve outcomes for clinical NLP research in epilepsy and other areas.

1 Introduction

Advances in Natural Language Processing (NLP), in particular pre-trained Transformers (Vaswani et al., 2017) and Large Language Models (LLMs), create opportunities to develop new methodologies to mine free-text Electronic Health Records (EHRs) for clinical research. One such opportunity is to investigate associations between anti-seizure medications (ASMs) and the frequency of seizures suffered by people with epilepsy, which is typically recorded in free text in the UK's National Health Service (NHS). Mostly, this text consists of doctors' and nurses' reports of outpatients' ambulatory visits; the reports are shared with a patient's primary care physician in the form of a letter. The majority of hospital care episodes for people with epilepsy occur in ambulatory care.

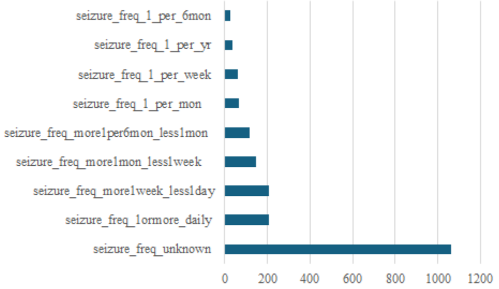


Figure 1: Distribution of 9 seizure frequency categories in annotated dataset.

Yet these reports are unstructured and typically noisy as they include a range of medical and administrative information, such as the patient's medication, other therapies, and details disclosed during previous clinic visits. Moreover, the reports often do not include any information about seizure frequency and, if they do, the language is often imprecise so that the nature of the frequency is vague or unclear. These factors make the application of NLP to EHRs to research seizure frequency challenging.

Epilepsy affects about 1% of the general population (Fiest et al., 2017). Around 30% of people with epilepsy do not respond to ASMs and are therefore regarded as refractory to treatment (Kwan and Brodie, 2000). While there are more than 30 individual ASMs and a much larger number of possible combinations of ASMs taken together, it is not feasible to try them in every refractory patient. This underlines the importance of research in predicting which ASMs would have the greatest impact on epileptic seizures for individual patients.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 526–535
August 16, 2024. ©2024 Association for Computational Linguistics

Although there is some published research on applying pre-trained Transformers to investigate epileptic seizure frequency among EHRs, the more recent opportunity of applying the new, generative LLMs for the same task is under-explored. However, it is expected that the application of generative LLMs to epilepsy research will increase significantly (van Diessen et al., 2024). The paucity of published research in this field may largely be due to the fact that these models are so new.

The most extensive relevant research we found was a long-term study (Xie et al., 2023; Xie et al., 2022a; and Xie et al., 2022b) that used a different methodology from ours to extract seizure frequency information from EHRs. The University of Pennsylvania researchers applied the pre-trained Transformers Bio_ClinicalBERT (for text classification), RoBERTa (for text extraction), and a T-5 model (to summarize sentences with seizure frequency data) to free-text data in EHRs to determine the seizure frequency of a person with epilepsy or whether that person was seizure free. For seizure frequency, they framed the task as an extractive question-answering problem, asking the language model to identify statements that answered the question: "How often does the patient have seizures?" They then simplified each sentence into a standardized format, "X per Y [day/month/year/visit]"; for example, "1 per 1 week". They subsequently manually annotated 1,000 sentences of seizure frequency generated by their models with the formatted summaries, then split them into training (700 sentences) and testing (300 sentences) datasets. Finally, they fine-tuned a T5-large model using Huggingface on the training dataset and made predictions on the test dataset. The researchers declared an "overall accuracy" score of 0.88 for seizure frequency, which comprised scores for each of "sentence accuracy", "summary accuracy", and "quantity accuracy".

That study follows a large body of research applying pre-trained Transformers to a wide variety of clinical tasks, such as predicting the risk of seizure recurrence among children with epilepsy (Beaulieu-Jones et al., 2023), inferring cancer disease response from free-text radiology reports (Tan et al., 2023), or detecting dementia with in-hospital clinical notes (Liu et al., 2023). Two other studies used rules-based NLP approaches to identify seizure frequency in unstructured clinic letters (Fonferko-Shadrach et al., 2019; Decker et al., 2022).

Our objective was to apply a new, generative LLM to the task of determining seizure frequency from free-text data. LLMs are built on the architecture of the Transformers but are much larger and more powerful language models. We were encouraged by recent research that demonstrates the benefits of using LLMs with clinical texts (for example, Agrawal et al., 2022; Thirunavukarasu et al., 2023; and Zhou et al., 2023). Our research, however, was restricted to using only an open-source language model because we used confidential NHS medical data that had to remain within the hospital's secure IT network for regulatory reasons. Therefore, we could not experiment with LLMs such as OpenAI's ChatGPT that are only available via an API to an off-site service. We found that Meta's Llama 2 (Touvron et al., 2023) performed best for our purposes within our limitations (see details in section 2.4). The LLM was run on up to eight Nvidia V100 GPUs.

2 Data and Methods

2.1 Data Collection

We selected 41,340 EHRs, the vast majority of which comprised doctors' and nurses' reports of outpatients' ambulatory visits, from King's College Hospital (KCH) in London spanning a decade from 2013-2022. The records related to 6,853 unique adult people with epilepsy being treated at KCH. We defined a person with epilepsy as someone who has at least one record of an epilepsy diagnosis. The selection was done via CogStack, an open-source information retrieval and extraction platform for EHRs developed by researchers at the NIHR Maudsley Biomedical Research Centre in London.[1] CogStack integrates with KCH's EHRs. We defined a set of epilepsy-related keywords and medical codes, and then used CogStack's search functionality to filter out EHRs that matched these definitions. We then used stratified random sampling to select 3,000 EHRs to create an annotated dataset, which ensured proportional distribution across the original dataset in regard to age, gender, and ethnicity to minimize bias. Subsequently, a team of six annotators, comprising four neuroscience clinicians and two data scientists, manually annotated the 3,000 EHRs for

[1] https://cogstack.org

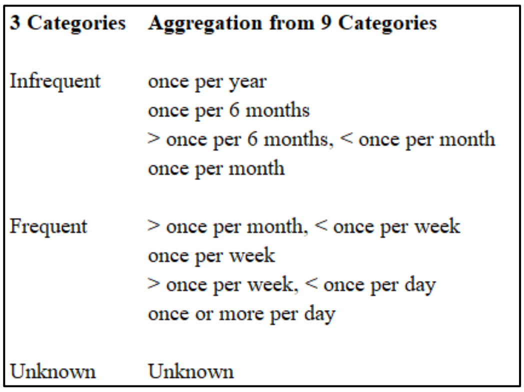
key data categories of the project, in particular seizure frequency, as well as seizure freedom, current anti-epilepsy medication, epilepsy type, seizure type, associated symptoms, and comorbidities. Due to time and resource limitations, as well as tight deadlines, the annotators worked on separate batches of the 3,000 EHRs, rather than having two annotators work on the same batch for moderation. A user guide was written for the annotators with instructions on how to annotate for each key data category, including seizure frequency with eight temporal frequencies and 'unknown' (see section 2.3 and Table 1 for more details).

2.2 Broader Research Project

This research on seizure frequency was part of a broader epilepsy research project run by the Department of Basic & Clinical Neuroscience in the School of Neuroscience at King's College London in the UK. The objective of the broader project is to apply machine learning at scale in an attempt to discover combinations of ASMs that enable refractory people with epilepsy to stop having seizures. Seizure frequency is a critical data point for this broader project.

2.3 Seizure Frequency Categories

We chose nine categories of seizure frequency for people with epilepsy, eight of which are for temporal frequencies and the last for unknown, meaning either the Electronic Health Record (EHR) contained no reference to seizures (which is common) or the LLM could not determine the frequency of seizures, due to the ambiguity of the text. We arrived at the nine categories after reviewing other studies (mostly non-NLP research) that investigated the frequency of epilepsy seizures (Wie et al., 2023; Westrhenen et al., 2022; Hsieh et al., 2022; Choi et al., 2014; and van Hout et al., 1997). Our aim was to stress test Llama 2 to gauge to what degree it could identify different seizure frequencies from unstructured text. We created shorthand labels for the nine seizure frequency categories for the annotation dataset (see Figure 1), mainly for ease-of-use when it came to writing Python code to evaluate the performance of Llama 2. Subsequently, we found that Llama 2 could often provide answers on the temporal duration of seizure frequency in an EHR in the format of our shorthand category labels following few-shot prompting instructions. However, after discovering that Llama 2 did not generate accurate enough predictions of seizure frequency over the nine categories, we then aggregated these nine categories into three categories (without performing a new experiment), which in turn resulted in Llama 2 predictions that were more accurate and usable for the broader epilepsy research project (see Table 1).



3 Categories	Aggregation from 9 Categories
Infrequent	once per year
	once per 6 months
	> once per 6 months, < once per month
	once per month
Frequent	> once per month, < once per week
	once per week
	> once per week, < once per day
	once or more per day
Unknown	Unknown

Table 1: 3 seizure frequency categories and aggregation from 9 categories.

2.4 Llama 2: Model Development and Implementation

We used LangChain as our development framework because it provides convenience and flexibility for building applications powered by LLMs.[2] First, we deployed LangChain in our local environment, then we downloaded a 13B parameter version of Llama 2 from Hugging Face and loaded it into LangChain.[3] LangChain offers simple interfaces for loading and initializing LLMs. After the model was loaded and initialized, we loaded various templates into LangChain, allowing us to perform multiple LLM operations in the local environment. While Meta provides 7B, 13B, and 70B different-sized models of Llama 2, our GPU platform did not have the computing power to run the largest 70B model. We used a chat version of Llama 2 13B that had been quantized by GPTQ. Although Meta released Llama 3 in April 2024, this did not provide enough time to run experiments using the latest Llama version in light of the submission deadline for this paper.

[2] https://www.langchain.com

[3] https://huggingface.co/TheBloke/Llama-2-13B-chat-GPTQ

Read the following context then work through these 3 steps.

1. Determine whether the context has any information about the frequency of the epilepsy patient's seizures.

2. If the context does not have any information about the frequency of the epilepsy patient's seizures, then you answer: ‘I do not know.’

3. If the context does have information about the frequency of the epilepsy patient's seizures, then you estimate the frequency of the epilepsy seizures and express the frequency in terms of per year, per month, per week, or per day, whichever is most relevant.

Figure 2: Query structure in 3 steps for Llama 2.

Clear example:

“We went through some of his seizures and in March he had two convulsions and three or four petit mal.”

Seizure diary example:

“Seizures: Partial seizures: July x 23, Aug x 0, Sept x 1, Oct so far x 7 (x1 daily 7th to 10th, 14th x1, 15th x 2, 18 x1.”

Ambiguous example:

“Louise and her mum confirm no seizures with her last seizure was possibly in November but they are not sure.”

Figure 3: Examples of context for epilepsy information in Electronic Health Records (excerpts from clinical letters**).**

As is well known with generative LLMs, the key issues with developing the model for seizure frequency extraction were prompt engineering and minimizing hallucinations. The problem of hallucinations – when LLMs generate plausible yet incorrect information – in clinical settings is explored at length in Pal et al., 2023, a study that found Llama 2’s 70B parameter model performed well in one of its tests. The free-text EHRs were passed without modification to the LLM.

We found that the generally accepted default setting for the temperature, 0.7, was too ‘creative’ for our purposes, encouraging Llama 2 to generate overly colorful answers to our seizure frequency questions and, on occasions, even providing diagnostic advice, including medication prescriptions, for the person with epilepsy. In turn, this increased the false positives. On the other hand, we concluded that a minimal temperature of 0.0001 was sufficient for the model to generate typically fact-based answers without excessive creativity and helped reduce the false positives.

Three aspects of prompt engineering proved critical for usable output. First, few-shot prompting significantly improved Llama 2’s ability to identify seizure frequency in an EHR, and proved much better than zero-shot prompting. However, we required 11 examples to give the model enough instructions on how to make complex decisions based on our nuanced nine categories of seizure frequency. Second, the characterization instructions in the prefix were a major factor in the model generating acceptable answers. Two key elements were instructing the model to act like a “professional neuroscientist who is responding to fellow neuroscientists” and to provide “succinct answers,” the latter helping to eliminate verbosity. Third, we discovered the query was optimally structured by asking the model to logically work through three numbered steps to determine seizure frequency, as distinct from asking a single question (see Figures 2 and 3).

During initial iterations, we experimented with query structures that involved simpler instructions without an explicit logical sequence or numbered steps. The selection process involved group discussion evaluating the model’s output from different variations of prompts, which in turn developed the optimal query structure. Of course, in the future improved prompt strategies and new LLMs may enhance the model outputs for extracting seizure frequency from EHRs, and this warrants further investigation.

The few-shot prompting examples provided Llama 2 with enough ‘education’ to generate answers that typically either matched, or closely resembled, our labels for the nine seizure frequency categories, thereby demonstrating the model’s ability to adapt its answers to idiosyncratic nomenclature. Of the 11 prompting examples, seven covered all but one of the temporal seizure frequency categories, two covered situations in which the patient did suffer seizures but the frequency of them was too difficult to determine from the EHR, and two covered situations in which the patient had not suffered seizures. We found during experimentation that doubling the last two kinds of prompts helped minimize hallucinations, or false positives. However, the model's answers were far from uniformly exact, as it often created its own versions of our category labels, so we devised an algorithm to interpret the model's answers if they either closely matched or were far from matching our labels. (See Appendix A for Llama 2 model architecture diagram.)

2.5 Annotation Dataset

The nine seizure frequency categories in the annotated dataset were dominated by unknowns, which comprised 71% of the 3,000 EHRs. In other words, only 29% of the annotated doctors' and nurses' reports contained any detectable information about seizure frequency. While some references to seizure frequency were clear and precise, especially if based on a patient's seizure diary, unfortunately many others were vague and imprecise. Consequently, the available data is sparse in regard to the core topic, which in turn makes the application of NLP to this task all the more challenging. Moreover, the number of observations in the higher frequency categories of seizure frequency – e.g., 'once or more per day' and 'more than once per week, less than once per day' – were roughly three times more common in the annotated dataset than those in the lower frequency categories (see Figure 1). This meant that Llama 2 found the lower frequency seizure categories more difficult to identify than the higher frequency categories.

2.6 BERT and RoBERTa: Model Development and Implementation

For a comparison to our Llama 2 method, we also fine-tuned BERT Large (Devlin et al., 2019) and RoBERTa Large (Liu et al., 2019) models on the annotated dataset, which was reduced from 3,000 EHRs to 1,720 EHRs to create a balanced dataset that was equally weighted between EHRs in which seizure frequency was known and EHRs in which seizure frequency was unknown. The unknown EHRs were reduced randomly to equal the 860 known EHRs. In turn, this reduced annotated dataset was restricted to the EHR text and the nine seizure frequency categories. Finally, it was split on an 80:10:10 ratio to create training, validation, and test datasets, respectively. We assume independent splits, a normal distribution, and a 95% confidence interval.

Both the BERT Large and RoBERTa models were used with PyTorch, an AdamW optimizer, threshold of 0.5 for the sigmoid, batch size of 4 (due to GPU memory limitations), and a learning rate of $1e^{-5}$. The optimal number of epochs varied for each model: 10 for BERT Large and 6 for RoBERTa Large. While the optimal dropout rate was 0.3 for BERT Large and 0.4 for RoBERTa Large. The maximum number of tokens for each EHR was set at 512, the upper limit for these two models.

3 Results

Our objective was to test an LLM against nine nuanced seizure frequency categories to determine how accurately they could identify seizure frequency from unstructured EHRs. The model F1 score for Llama 2 on the full annotated dataset of 3,000 EHRs was 0.73 and the model accuracy 0.94 (see Table 3), although the accuracy figure is misleading because it is boosted by a high number of true negatives, hence we prefer F1 as a measure of performance. We found that Llama 2 did well in identifying letters that had no or ambiguous information about seizure frequency, recording an F1 score of 0.87, and did moderately well on the most common known categories ('more than once a week', 0.35; and 'one or more daily', 0.41). But Llama 2 struggled with the remaining six temporal categories, ranging from 'once a week' to 'once a year' (see Table 2). Therefore, we aggregated the nine seizure frequency categories into three categories (infrequent, frequent, and unknown) to improve the performance of the model (see Table 4). Under the three categories, Llama 2 posted F1 scores of 0.87 for the unknowns, 0.62 for frequent seizures, and a lower 0.30 for infrequent seizures. Results are the average of three different runs of Llama 2. The LLM's output was highly consistent on each run, reflecting the low temperature of 0.0001 that in turn minimizes 'creativity' in answers.

	Seizure Frequency 9 Categories: F1 Score								
			> once / 6 months		> once / month		> once / week		
Model	Once / year	Once / 6 months	< once / month	Once / month	< once / week	Once / week	< once / day	1 or more / day	Unknown
LlaMA 2 13B	0.11	0.06	0.17	0.42	0.36	0.06	0.35	0.41	0.87
RoBERTa Large	0.00	0.00	0.61	0.00	0.63	0.00	0.48	0.59	0.74
BERT Large	0.00	0.00	0.47	0.00	0.36	0.00	0.55	0.58	0.76

Table 2: Model performance evaluation on 9 seizure frequency categories.

	Model	Model
Model	F1 Score	Accuracy
LlaMA 2 13B	0.73	0.94
RoBERTa Large	0.58	0.91
BERT Large	0.55	0.90

Table 3: Model performance for F1 and accuracy

	Seizure Frequency 3 Categories: F1 Score		
Model	Infrequent	Frequent	Unknown
LlaMA 2 13B	0.30	0.62	0.87
RoBERTa Large	0.43	0.76	0.74
BERT Large	0.39	0.77	0.76

Table 4: Model performance evaluation on 3 seizure frequency categories.

By comparison, Llama 2 performed better than BERT Large and RoBERTa Large, although it must be noted that our testing methodology for Llama 2 was different to that for the pre-trained Transformers. Llama 2 was mainly tested against the full annotated dataset of 3,000 EHRs (Llama 2 does not require fine-tuning), whereas BERT Large and RoBERTa Large, which required 80% of the balanced annotated dataset of 1,720 EHRs as a training dataset, were tested against a much smaller test dataset of 172 EHRs (or 10% of 1,720). Under this scenario, Llama 2's model F1 score of 0.73 was higher than RoBERTa Large's 0.58 and BERT Large's 0.55 (the results of the pre-trained Transformers are the average of three different runs with the same random states). Moreover, Llama 2 recorded a positive F1 score in all nine seizure frequency categories, whereas the pre-trained Transformers both posted F1 scores of zero in at least four categories, suggesting Llama 2 is better at identifying seizure frequency in the sparse categories.

However, we also tested Llama 2's performance on the same smaller test dataset of 172 EHRs used for BERT Large and RoBERTa Large. In this case, Llama 2's model F1 score dropped to 0.54, broadly in line with the pre-trained Transformers, and Llama 2 recorded F1 scores of zero in three of the nine seizure frequency categories. There are two possible explanations for this apparent difference in performance. First, the small test dataset represents only 6% of the full annotated dataset of 3,000 EHRs, therefore the latter is a better guide of actual model performance. Second, only 60% of EHRs in the small test dataset contained data on seizure frequency, and of those EHRs there was very little data on four categories ('once per week', 'once per month', 'once per six months', 'once per year'), therefore the paucity of data in the less common categories presented a greater challenge for the few-shot prompting structure for the LLM.

Furthermore, other metrics demonstrate that even when evaluated on the small test dataset, Llama 2 was more reliable than the other two models. Llama 2 predicted that 59% of the EHRs in the test dataset contained either no, or vague, information about seizure frequency, which was higher than the annotators' 40% but lower than RoBERTa Large's 65% and BERT Large's 71%. Also, while Llama 2 always made a prediction on every EHR, RoBERTa Large failed to make a prediction on average on 17% of the test EHRs and BERT Large failed on 30%.

It is difficult to compare the results of our study to those of Xie et al. (2023, 2022a, and 2022b) because they provided an "overall accuracy" score of 0.88 for seizure frequency and did not break down accuracy for individual seizure frequency categories. However, in broad terms in appears our Llama 2 methodology produced at least similar performance given its model accuracy was 0.94

and its accuracy rate for the infrequent category was 0.92 and for the frequent category 0.85.

4 Discussion

While our initial aim to determine whether the LLM could identify the frequency of seizures in unstructured outpatient reports for eight temporal categories proved too ambitious, when the temporal categories were reduced to frequent and infrequent the output of Llama 2 was much improved. The key objective of our broader epilepsy project is to track the effects of different combinations of anti-seizure medications on seizure frequency in individual patients and consequent changes. In this respect, Llama 2's F1 scores of 0.87 for the unknowns and 0.62 for frequent seizures is useful. Although the model's F1 score was a lower 0.30 for infrequent seizures, we are mindful that the number of observations of frequent seizures is roughly three times that of infrequent seizures, as previously stated, and so while more work is required to optimize the model's output for infrequent seizures, its overall performance aids our broader objective.

During experimentation it was clear that Llama 2's pre-training on vast general corpora had imbued it with a noticeable degree of expert knowledge about epilepsy. This may be one reason why Llama 2 proved superior to the pre-trained Transformers in identifying seizure frequency in unstructured, free-text EHRs. Another reason is that Llama 2 is a much bigger language model – we used the 13B parameter version – than BERT Large with 336M parameters and RoBERTa Large with 356M parameters.

Llama 2, like other generative LLMs, has three advantages over pre-trained Transformer language models. First, Llama 2 does not have to be fine-tuned on an annotated dataset, which saves substantial time and resources by obviating annotations for a training dataset. Second, Llama 2 does not have a built-in maximum token length for processing long texts. Third, Llama 2 is 'guided' on a particular language task by prompt engineering, which typically takes less time than adjusting multiple hyperparameters to optimize the performance of a pre-trained Transformer model.

On the other hand, Llama 2 has a distinct disadvantage: because of its large size, the LLM requires a longer running time. In this case, Llama 2 took on average 3.6 seconds to process one EHR, or about one hour for 1,000 EHRs.

A drawback of this particular study, however, is that the results are not reproduceable by other researchers because the patient EHRs are confidential and can only be accessed via the hospital's secure IT network.

5 Conclusion

Llama 2, as a popular LLM widely regarded as producing impressive performance on a variety of NLP tasks, performed well on the specific clinical NLP task of identifying seizure frequency from unstructured, free-text EHRs. This demonstrates that the new, generative LLMs are useful for epilepsy research in particular and clinical NLP research in general. The key question for our broader epilepsy research project was whether a new, generative LLM could identify seizure frequency among the EHRs to a sufficient degree to use the model's predictions as a basis for further research into different anti-epilepsy medications and their effects on seizure frequency. Our conclusion is that Llama 2 can.

Limitations

The confidential nature of the EHRs creates two limitations of this study. First, the model outputs are not reproduceable by research teams outside the hospital where the authors worked because the data has to remain within the hospital's secure IT network for regulatory reasons. Second, we could not experiment with LLMs such as OpenAI's ChatGPT that are only available via an API to an off-site service due to privacy reasons. However, with more time we could have experimented with other open-source LLMs. Another limitation is that, because of time and resource constraints, our annotation methodology of having six expert annotators working on separate batches of the 3,000, rather than having two annotators work on the same batch for moderation, did not allow for a measurement of inter-annotator agreement. Also, our research was also limited by the computing power generated by our GPU platform (eight Nvidia V100 GPUs). For example, this did not have the capacity to work with Llama 2's 70B parameter version on our dataset. Finally, the dataset of epilepsy patients from King's College Hospital may differ from datasets of epilepsy patients from other hospitals.

Ethical Considerations

The main ethical consideration was that the confidential EHRs of patients had to remain within the hospital's secure IT network. Therefore, researchers could only access the data and ingest the data into models via the hospital's IT network. Researchers and clinicians required clearance from the hospital. The project operated under London 593 South-East Research Ethics Committee (reference 18/LO/2048), approval granted to the King's 595 Electronic Records Research Interface (KERRI).

Acknowledgements

This work was supported by Epilepsy Research Institute UK (project reference 2209 Richardson Angelini) in conjunction with Angelini Pharma. The authors also wish to thank Angus Roberts and Joe Davies for their advice on early drafts, as well as the reviewers for the BioNLP Workshop for their constructive and helpful feedback.

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 1998-2022, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Brett K. Beaulieu-Jones, Mauricio F Villamar, Phil Scordis, Ana Paula Bartmann, Waqar Ali, Benjamin D Wissel, Emily Alsentzer, Johann de Jong, Arijit Patra, Prof Isaac Kohane. 2023. Predicting seizure recurrence after an initial seizure-like episode from routine clinical notes using large language models: a retrospective cohort study. *The Lancet Digital Health*, vol. 5, issue 12: pages e882-94.

Iz Beltagy, Matthew E. Peters, Arman Cohan. 2020. Longformer: The Long-Document Transformer. arXiv:2004.05150v2. Version 2.

Hyunmi Choi, Marla J. Hamberger, Heidi Munger Clary, Rebecca Loeb, Frankline M. Onchiri, GusBaker, W. Allen Hauser, and John B. Wong. 2014. Seizure frequency and patient-centered outcome assessment in epilepsy. *Epilepsia* 55(8): pages 1205-1212.

Barbara M. Decker, Alexandra Turco, Jian Xu, Samuel W. Terman, Nikitha Kosaraju, Alisha Jamil, Kathryn A. Davis, Brian Litt, Colin A. Ellis, Pouya Khankhanian, Chloe E. Hill. Development of a natural language processing algorithm to extract seizure types and frequencies from the electronic health record. *Seizure: European Journal of Epilepsy* 101 (2022): 48-51. doi.org/10.1016/j.seizure.2022.07.010

Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova (Google AI Language). 2019. BERT: Pretraining of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805v2. Version 2.

Eric van Diessen, Ramon A. van Amerongen, Maeike Zijlmans, Willem M. Otte. 2024. Potential merits and flaws of large language models in epilepsy care: A critical review. *Epilepsia* 00: pages 1-14. https://doi.org/10.1111/epi.17907.

Kirsten M. Fiest, Khara M. Sauro, Samuel Wiebe, Scott B. Patten, Churl-Su Kwon, Jonathan Dykeman, Tamara Pringsheim, Diane L. Lorenzetti, Nathalie Jetté. 2017. Prevalence and incidence of epilepsy: A systematic review and meta-analysis of international studies. *Neurology* 88(3): pages 296-303.

Beata Fonferko-Shadrach, Arron S Lacey, Angus Roberts, Ashley Akbari, Simon Thompson, David V Ford, Ronan A Lyons, Mark I Rees, William Owen Pickrell. Using natural language processing to extract structured epilepsy data from unstructured clinic letters: development and validation of the ExECT (extraction of epilepsy clinical text) system. *BMJ* Open. 2019 Apr 1; 9(4): e023232. doi: 10.1136/bmjopen-2018-023232.

Ben van Hout, Dennis Gagnon, Eric Souetre, Sibylle Ried, Claude Remy, Gus Baker, Pierre Genton, Herve Vespignani, and Pauline McNulty. 1997. Relationship Between Seizure Frequency and Costs and Quality of Life of Outpatients with Partial Epilepsy in France, Germany, and the United Kingdom. *Epilepsia* 38(11): pages 1221-1226.

Jason K. Hsieh, Francesco G. Pucci1, Swetha J. Sundar, Efstathios Kondylis, Akshay Sharma, Shehryar R. Sheikh, Deborah Vegh, Ahsan N. Moosa, Ajay Gupta, Imad Najm, Richard Rammo, William Bingaman, Lara Jehi. 2022. Beyond seizure freedom: Dissecting long- term seizure control after surgical resection for drug-resistant epilepsy. *Epilepsia*, vol. 64: pages 103-113.

P. Kwan and M.J. Brodie. 2000. Early identification of refractory epilepsy. *The New England Journal of Medicine* 342(5): pages 314-9.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692.

Ming Liu, Richard Beare, Taya Collyer, Nadine Andrew, and Velandai Srikanth. 2023. Leveraging Natural Language Processing and Clinical Notes for Dementia Detection. In Proceedings of the 5th Clinical Natural Language Processing Workshop, pages 150-155, Toronto, Canada. Association for Computational Linguistics.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2023. Med-HALT: Medical Domain Hallucination Test for Large Language Models. In Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL), pages 314–334, Singapore. Association for Computational Linguistics.

Ryan Shea Ying Cong Tan, Qian Lin, Guat Hwa Low, Ruixi Lin, Tzer Chew Goh, Christopher Chu En Chang, Fung Fung Lee, Wei Yin Chan, Wei Chong Tan, Han Jieh Tey, Fun LoonLeong, Hong Qi Tan, WenLong Nei, WenYeeChay, David WaiMeng Tai, Gillianne Geet Yi Lai, Lionel Tim-Ee Cheng, Fuh Yong Wong, Matthew Chin Heng Chua, Melvin Lee Kiang Chua, Daniel Shao Weng Tan, Choon Hua Thng, Iain Bee Huat Tan, and HweeTouNg. 2023. Inferring cancer disease response from radiology reports using large language models with data augmentation and prompting. *Journal of the American Medical Informatics Association*, 30(10): pages 1657-1664.

Arun Thirunavukarasu, Kabilan Elangovan, Darren Shu Jeng Ting, Laura Gutierrez, Daniel Shu Wei Ting. 2023. Large language models in medicine. *Nature Medicine*, vol. 29: pages 1930-1940.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, Thomas Scialom. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288v2. Version 2.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. arXiv:1706.03762v5. Version 5.

Anouk van Westrhenen1, Ben F. M. Wijnen, Roland D. Thijs. 2022. Parental preferences for seizure detection devices: a discrete choice experiment. *Epilepsia*, vol. 63: pages 1152-1163.

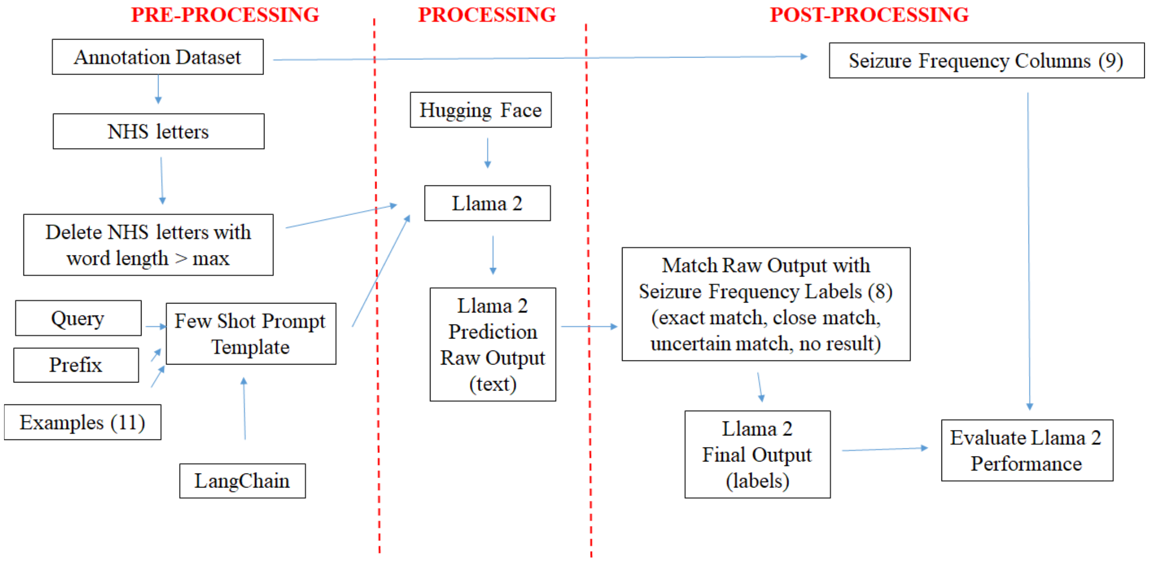
Kevin Xie, Brian Litt, Dan Roth, and Colin A. Ellis. 2022a. Quantifying Clinical Outcome Measures in Patients with Epilepsy Using the Electronic Health Record. In Proceedings of the 21st Workshop on Biomedical Language Processing, pages 369-375, Dublin, Ireland. Association for Computational Linguistics.

Kevin Xie, Ryan S. Gallagher, Erin C. Conrad, Chadric O. Garrick, Steven N. Baldassano, John M. Bernabei, Peter D. Galer, Nina J. Ghosn, Adam S. Greenblatt, Tara Jennings, Alana Kornspun, Catherine V. Kulick-Soper, Jal M. Panchal, Akash R. Pattnaik, Brittany Scheid, Danmeng Wei, Micah Weitzman, Ramya Muthukrishnan, Joongwon Kim, Brian Litt, Colin Ellis, Dan Roth. 2022b. Extracting seizure frequency from epilepsy clinic notes: a machine reading approach to natural language processing. *Journal of the American Medical Informatics Association*, 29(5): pages 873-881.

Kevin Xie, Ryan S. Gallagher, Russell T. Shinohara, Sharon X. Xie, Chloe E. Hill, Erin C. Conrad, Kathryn A. Davis, Dan Roth, Brian Litt, Colin A. Ellis. 2023. Long-term epilepsy outcome dynamics revealed by natural language processing of clinic notes. *Epilepsia* 64(7): pages 1900-1909.

Weipeng Zhou, Majid Afshar, Dmitriy Dligach, Yanjun Gao, and Timothy Miller. 2023. Improving the Transferability of Clinical Note Section Classification Models with BERT and Large Language Model Ensembles. In Proceedings of the 5th Clinical Natural Language Processing Workshop, pages 125–130, Toronto, Canada. Association for Computational Linguistics.

Appendix A. Model Architecture Diagram for Llama 2


PRE-PROCESSING
PROCESSING
POST-PROCESSING
Annotation Dataset
NHS letters
Delete NHS letters with word length > max
Query
Prefix
Examples (11)
Few Shot Prompt Template
LangChain
Hugging Face
Llama 2
Llama 2 Prediction Raw Output (text)
Seizure Frequency Columns (9)
Match Raw Output with Seizure Frequency Labels (8) (exact match, close match, uncertain match, no result)
Llama 2 Final Output (labels)
Evaluate Llama 2 Performance

How do you know that? Teaching Generative Language Models to Reference Answers to Biomedical Questions

Bojana Bašaragin
The Institute for AI of Serbia
Fruškogorska 1, Novi Sad
Serbia
bojana.basaragin@ivi.ac.rs

Adela Ljajić
The Institute for AI of Serbia
Fruškogorska 1, Novi Sad
Serbia
adela.ljajic@ivi.ac.rs

Darija Medvecki
The Institute for AI of Serbia
Fruškogorska 1, Novi Sad
Serbia
darija.medvecki@ivi.ac.rs

Lorenzo Cassano
Bayer A.G.
Müllerstraße 178, Berlin
Germany
lorenzo.cassano@bayer.com

Miloš Košprdić
The Institute for AI of Serbia
Fruškogorska 1, Novi Sad
Serbia
milos.kosprdic@ivi.ac.rs

Nikola Milošević
Bayer A.G.
Müllerstraße 178, Berlin
Germany
nikola.milosevic@bayer.com

Abstract

Large language models (LLMs) have recently become the leading source of answers for users' questions online. Despite their ability to offer eloquent answers, their accuracy and reliability can pose a significant challenge. This is especially true for sensitive domains such as biomedicine, where there is a higher need for factually correct answers. This paper introduces a biomedical retrieval-augmented generation (RAG) system designed to enhance the reliability of generated responses. The system is based on a fine-tuned LLM for the referenced question-answering, where retrieved relevant abstracts from PubMed are passed to LLM's context as input through a prompt. Its output is an answer based on PubMed abstracts, where each statement is referenced accordingly, allowing the users to verify the answer. Our retrieval system achieves an absolute improvement of 23% compared to the PubMed search engine. Based on the manual evaluation on a small sample, our fine-tuned LLM component achieves comparable results to GPT-4 Turbo in referencing relevant abstracts. We make the dataset used to fine-tune the models and the fine-tuned models based on Mistral-7B-instruct-v0.1 and v0.2 publicly available.

1 Introduction

The idea of automated referencing dates back to 1970 when (Garfield, 1970) proposed an automatic system where a computer evaluates the appropriateness of references within an article. With the emergence of generative large language models (LLMs), numerous systems are being developed to answer specific questions, supported by relevant references (Huang and Chang, 2024; Menick et al., 2022; Yang et al., 2023). Generative LLMs can produce answers that appear coherent, confident and articulate. However, the information conveyed may not be correct or verifiable. Furthermore, the limited internal knowledge of generative LLMs can hinder their ability to deliver factually accurate answers, particularly within specialized fields (Gravel et al., 2023; Zheng et al., 2023). This issue is notably concerning in the biomedical domain, where accurate and factual answers are critical. The scientific community has recognized the dangers of factually incorrect or nonsensical information and has been reluctant to utilize these models to their potential. Providing an opportunity for scientists to obtain correct and verifiable answers to questions is an opportunity to increase scientific productivity and its impact. Moreover, privacy, sovereignty and security concerns in pharma and biomedicine often necessitate building systems where all components are controllable (e.g. deployed in-house), to avoid reliance on third-party APIs such as OpenAI[1], especially when secret data is concerned.

Incorporating domain-specific external knowledge beyond LLM data is essential for mitigating hallucinations in LLMs. The retrieval-augmented generation (RAG) approach, which integrates the generative capabilities of an LLM with a specialized retrieval system, enhances the model's accuracy and relevance by grounding its responses in verified information.

In this paper, we present a biomedical RAG system consisting of a hybrid search based on

[1] https://openai.com

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 536–547
August 16, 2024. ©2024 Association for Computational Linguistics

PubMed[2] and fine-tuned generative models for referenced question-answering (QA). We make both the models and the dataset used to fine-tune the models publicly available.

The remainder of this paper is organized as follows: Section 2 provides a review of related work on reliability and verifiability of the LLM generated content and the approaches to generating texts with references. Section 3 describes the design of the IR and generative components. We evaluate the components in Section 4, first individually and then jointly. We end the paper with conclusions and some future work remarks in Section 5.

2 Related work

Generative LLMs, such as GPT and similar architectures, have enabled question-answering (QA) tasks across various domains, including medicine. The current state of these models is characterized by several challenges, particularly regarding the verifiability and reliability of the information they generate. By evaluating ChatGPT responses and references in the medical domain, (Gravel et al., 2023) found that 69% of generated references were fabricated, while professionals rated the answers at a median quality of 60%. Similarly, when (Liu et al., 2023) conducted manual evaluations of four prominent generative search engines Bing Chat, NeevaAI, perplexity.ai, and YouChat, they found that while the responses of these engines were fluent and seemingly informative, only 51.5% of sentences generated by these engines were fully supported by their citations, and merely 74.5% of citations accurately supported the statements they were linked to. These results leave space for improvement.

In general, there are two approaches to generating text with references (Huang and Chang, 2024). The first assumes training LLMs to produce references from parametric knowledge (information internalized from the training data). The second one assumes producing references from non-parametric knowledge (content retrieved from external sources).

The first approach, integrating citations directly from LLM's parametric knowledge, poses a significant technical challenge. Unlike search engines and IR systems that rely on indices for data retrieval, LLMs encode information into hidden representations during training, lacking a direct index. Therefore, referencing the sources of information becomes intricate. Despite these challenges, approaches have been suggested to train LLMs to include references using source identifiers (Taylor et al., 2022). However, these methods exhibit certain limitations, including citation inaccuracies and being restricted to academic citations.

The second approach, known as retrieval-augmented generation (RAG), combines generative LLMs with IR systems to form a hybrid system (Lewis et al., 2020). Here, the model is trained to recognize instances requiring citations, and the IR system retrieves suitable sources to provide context to the LLM. As a result, the LLM incorporates these sources as citations into its outputs, improving the credibility and accuracy of responses. While pre-trained and fine-tuned LLMs rely solely on their parametric knowledge, RAG integrates a customized external knowledge base without additional training, thus reducing hallucinations. Moreover, annotators often perceive RAG-enhanced answers to be more factual and specific compared to those from fine-tuned models (Lewis et al., 2020).

3 Method

The RAG system we propose in this paper is designed to perform referenced QA in the biomedical domain. It consists of two main components. The IR component, based on hybrid semantic and lexical search, retrieves relevant PubMed abstracts and provides a context for the generative LLM. The final system output is an answer to the user query, which contains a reference for each of the claims extracted from the relevant abstracts. The overview of the system architecture can be seen in Figure 1.

3.1 Information Retrieval Component

Our IR component uses data from PubMed database[3] containing citations and biomedical literature from several literature resources. The IR system integrates both sparse vectors (lexical index) and dense vectors (semantic index), enabling lexical and semantic search, and a hybrid combination of the two.

For the lexical retrieval, based on a ranking function Best Matching 25 (BM25), we use the OpenSearch[4] to create an index for PubMed articles, by concatenation of title and abstract as an indexed field. Also, we add authors' names, pub-

[2] https://pubmed.ncbi.nlm.nih.gov

[3] https://pubmed.ncbi.nlm.nih.gov/download/

[4] https://opensearch.org/

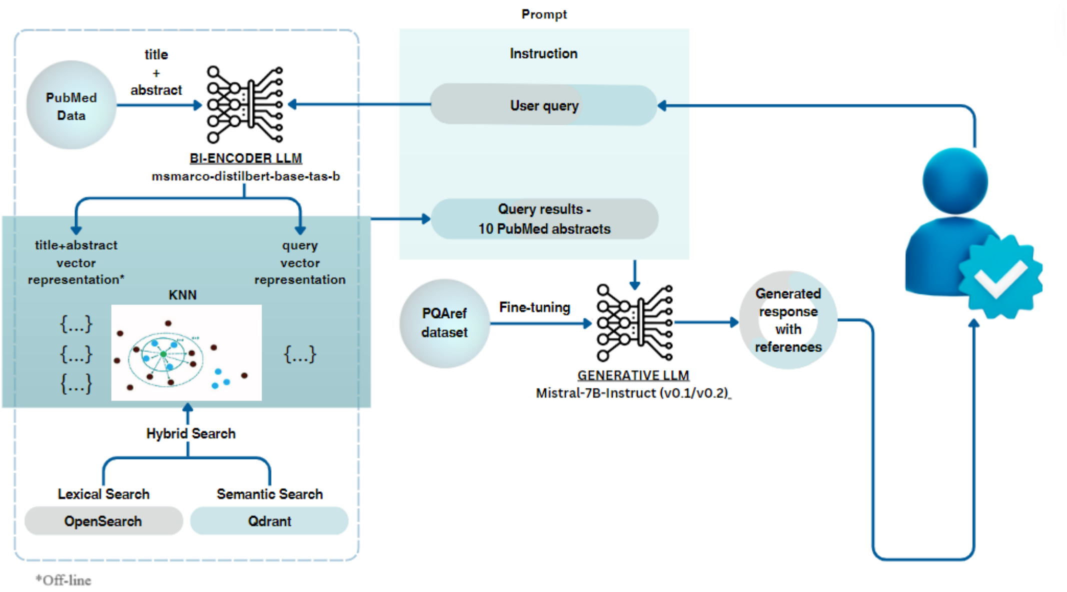


Figure 1: Architecture of our RAG system.

lication dates, and journal names as metadata for filtering.

For semantic retrieval, based on dense vectors, we use the Qdrant[5] vector database. Qdrant allowed the usage of memory mapping of vectors to a hard drive, reducing the memory (RAM) requirements of the system. To optimize semantic search retrieval time, we used 8-bit quantized embeddings, with the option to use full embeddings for rescoring the results.

We use the Hierarchical Navigable Small World (HNSW) indexing technique for Approximate Nearest Neighbors with dot product metrics to perform vector comparisons (Malkov and Yashunin, 2018). To create vector embeddings we use a bi-encoder sentence transformer model pre-trained on the MSMarco dataset (Hofstätter et al., 2021), which, at the time of indexing, had the best performance on Passage Retrieval Task[6].

In a corpus of 36,797,469 abstracts, 11,308,679 were found to be empty and thus omitted from the index. These empty abstracts predominantly originate from articles published in the pre-digital era, articles from journals that are not accessible for free, or journals that do not require abstracts. After eliminating these empty abstracts, we constructed two indices in the offline mode, designed for subsequent use in online semantic and lexical searches. The lexical index is created by indexing concatenated fields of titles and abstracts along with additional fields from PubMed articles for filtering purposes. The process of generating embeddings for the semantic index includes the creation of embeddings for titles and abstract concatenation using the model. This process is depicted in Figure 1, marked with an asterisk. Before generating embeddings for semantic search, it was ascertained that the average number of tokens within the dataset's title and abstract concatenation was 650. Given that the maximum input size of the model employed for embedding creation is 512 tokens, abstracts exceeding this threshold were subdivided into segments each containing no more than 512 tokens, and were indexed separately. The split was made at the end of the sentence before the 512th token.

In our case, hybrid search is a combination of lexical and semantic IR components. To utilize the hybrid search, we normalized scores from these two IR methods to scales ranging from 0 to 1. The scores from each of the search methods are then multiplied by the importance weights for each of the methods. This allows both the identification of direct matches and greatly improves the ability to discover semantically related phrases and text segments, even in the absence of exact textual matches.

3.2 Generative Component

The generative component of our system is based on the Mistral-7B model. Despite having fewer parameters, Mistral-7B shows superior performance

[5] https://qdrant.tech/

[6] https://www.sbert.net/docs/pretrained-models/msmarco-v3.html

over larger models such as Llama 2 13B across all evaluated benchmarks and Llama 1 34B in reasoning benchmarks, maths, and code generation (Jiang et al., 2023). Compared to its 0.1 version, Mistral-7B v0.2 introduced an expanded context window (32K to the previous 8K) and several other adjustments (rope-theta = 1e6, no sliding-window attention) contributing to more accurate and consistent outputs, improved efficiency, and adaptability to many different tasks (Anakin.ai, 2024).

For the sake of comparison, we opted for testing both currently available instruction-tuned versions of Mistral-7B (v0.1[7] and v0.2[8]). We test both models in the zero-shot mode but also fine-tune them using a custom dataset for referenced QA (see Section 3.2.1).

The input for the generative component consists of a user query and 10 abstracts retrieved by the IR component as most relevant for the user query. While generating the answer, the models perform another relevance check and answer the question using only the abstracts they find relevant. The final output is a concise answer that contains an abstract ID as a reference after each claim originating from the 10 abstracts.

In the following subsections, we briefly describe the dataset we used to fine-tune these models, as well as the fine-tuning process.

3.2.1 Dataset

We created a custom dataset to fine-tune the LLMs for the task of referenced QA. The dataset consists of 9075 questions, where each question is followed by 10 relevant abstracts (along with titles and PMIDs) and referenced answers to the questions based on the provided abstracts.

The questions were randomly selected from the PubMedQA dataset (Jin et al., 2019). The most relevant abstracts for each of these questions were retrieved from the PubMed repository using a combination of entity and free text search. To create the answers based on the retrieved abstracts, we used GPT-4 Turbo, specifically gpt-4-1106-preview[9], a GPT-4 Turbo preview model featuring improved instruction following. GPT-4 Turbo is currently the number one model on the Chatbot Arena leaderboard, a crowdsourced open platform for LLM evaluation (Chiang et al., 2024). The prompt we used to instruct GPT-4 Turbo to use references (PMIDs) was as follows:

> Answer the question using relevant abstracts provided, up to 300 words. Reference the statements with the provided abstract_id in brackets next to the statement.

To ensure the completeness of answers, GPT-4 Turbo was further instructed to continue generating if there is more content to generate. The answers were then automatically checked for completeness and incomplete final sentences were removed, which finally led to the size of answers ranging from 69 to 1221 tokens. In a small number of cases (25 questions) there was no direct answer in the abstracts so the answer does not contain any references. The total input length in the dataset (question + abstracts + answer) ranges from 1686 to 6987 tokens.

We name this dataset PQAref and make it available through Hugging Face[10].

3.2.2 Fine-tuning the models

Both Mistral-7B instruction-tuned versions were fine-tuned for the task of referenced QA using the QLoRA methodology (Dettmers et al.), allowing us to fine-tune the models on a single DGX NVIDIA A100-40GB GPU in ∼32 hours. The parameters we used for both models were standard loss, rank of 64, alpha of 16, and LoRA dropout of 0.1, resulting in 27,262,976 trainable parameters in both cases. Both models were fine-tuned over 2 epochs, using a batch size of 1. The PQAref dataset split was 80:10:10, with most inputs in the size range of 4000 to 6000 tokens in all three splits (see Figure 2).

We make the QLoRA adapters for both models available on Hugging Face as Mistral-7B-Instruct-v0.1-pqa-10[11] and Mistral-7B-Instruct-v0.2-pqa-10[12].

4 Results

4.1 Evaluation of IR Component

To evaluate our IR system, we utilized the BioASQ dataset (BioASQ team, 2024). The BioASQ dataset

[7] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.1

[8] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2

[9] https://platform.openai.com/docs/models/gpt-4-turbo-and-gpt-4

[10] https://huggingface.co/datasets/BojanaBas/PQAref

[11] https://huggingface.co/BojanaBas/Mistral-7B-Instruct-v0.1-pqa-10

[12] https://huggingface.co/BojanaBas/Mistral-7B-Instruct-v0.2-pqa-10

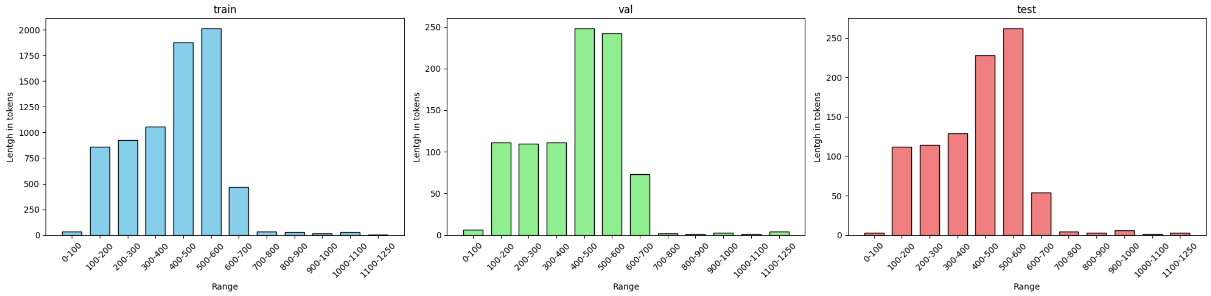


Figure 2: Distribution of answer length across train, val and test splits.

is designed for tasks that help drive advancements in biomedical information retrieval and QA. It includes 5049 questions along with corresponding gold-standard answers, relevant document snippets, and the PubMed IDs (PMIDs) of articles that are relevant to each question.

We compared the PMIDs retrieved by our system against the gold-standard PMIDs provided in the BioASQ dataset. This comparison was quantified using the precision metric, measuring the proportion of relevant identifiers retrieved by our system out of the total PMIDs retrieved. We evaluate precision using 10 retrieved documents (P@10) and mean average precision for 10 retrieved documents (MAP@10). The evaluation of the retrieval component is done using: (1) only lexical, (2) only semantic, and (3) a combination of the two. Additionally, we experimented with different weights for the lexical and semantic combinations.

For the lexical search, we experimented with stopword removal from the query and obtained better results compared to lexical search without stopword removal as shown in Table 1.

For semantic search, we experimented with three approaches: semantic search with full embeddings, semantic search with compressed embeddings (using 8-bit quantization), and semantic search using compressed embeddings with rescoring (using full embeddings for rescoring).

Semantic search with full embeddings had an average response time of 30 seconds, making it inefficient and unusable for real-world applications.

For semantic search with rescoring, we used compressed embeddings to retrieve 100 results, then rescored the top 10 using full-size embeddings. This method improved precision by 0.3% and was only 52 milliseconds slower than the approach without rescoring (see rows 1 and 2 in Table 1). Given the minimal additional time required, we tested the various weight combinations of hybrid search incorporating semantic search with rescoring. Parallel execution of semantic and lexical search further contributes to the time efficacy of the system (as shown in Table 1), reducing the average execution time from 489ms to 442ms.

Table 1: Our IR and PubMed search engine performance evaluation on the BioASQ dataset.

	P@10	MAP@10	time [ms]
1. Semantic without rescore	14.0%	25.7%	245
2. Semantic with rescore	14.4%	26.0%	297
3. Hybrid with rescore (lex. 0.1 sem. 0.9)	24.7%	32.5%	442
4. Hybrid with rescore (lex. 0.2 sem. 0.8)	24.7%	32.5%	442
5. Hybrid with rescore (lex. 0.3 sem. 0.7)	24.7%	32.5%	442
6. Hybrid with rescore (lex. 0.4 sem. 0.6)	24.7%	32.6%	442
7. Hybrid with rescore (lex. 0.5 sem. 0.5)	25.2%	41.0%	442
8. Hybrid with rescore (lex. 0.6 sem. 0.4)	30.7%	42.0%	442
9. Hybrid with rescore (lex. 0.7 sem. 0.3)	**30.8%**	**42.5%**	**442**
10. Hybrid with rescore (lex. 0.8 sem. 0.2)	30.8%	42.5%	442
11. Hybrid with rescore (lex. 0.9 sem. 0.1)	30.8%	42.6%	442
12. Lexical with stopwords removal	28.7%	41.1%	189
13. Lexical without stopwords removal	28.3%	40.1%	189
14. PubMed without MeSH Terms	9.2%	15.3%	698
15. PubMed with MeSH Terms	12.0%	19.1%	742

From the experiments detailed in Table 1, it is evident that the performance of semantic search alone is suboptimal, with notable enhancements observed upon integration with lexical search. The initial improvement is noted with the hybrid search employing a 0.1 lexical search weight, followed by a second significant enhancement achieved with a 0.6 lexical search weight (yielding absolute improvements of 10.3% and 16.3% respectively). Increasing the lexical search weight beyond 0.6 does not yield noticeably different outcomes. Assigning a weight of 1 to lexical search in hybrid search excludes semantic search, effectively reducing the system to pure lexical search, which produces worse results.

As the subsequent generative component does not account for the order of retrieved documents, we employ the P@10 metric to determine the most effective combination of parameters for hybrid search. After evaluating various configurations, we identified the optimal parameters for hybrid search:

a lexical search weight of 0.7 and a semantic component weight of 0.3. By allocating a higher weight to the semantic search component (0.3 in row 9 instead of 0.1 in row 11), we enhance the model's ability to capture and utilize the deeper, contextual relationships inherent in biomedical texts. Consequently, as shown in row 9, we choose these parameter values to conduct a hybrid search in our system.

Additionally, we evaluated the performance of PubMed search engine on the BioASQ dataset and got the P@10 of 9.2% and MAP@10 of 15,3% when searching without MeSH terms and P@10 of 12% and MAP@10 of 19.1% when searching with MeSH terms (rows 14 and 15 in Table 1).

4.2 Evaluation of the Generative Component

For the purpose of standalone evaluation of the generative component, we use the PQAref test set. We conducted automated and manual evaluations for the task of referenced QA, which involved analyzing the total number of all references per answer and relevant references per answer, checking the correctness of IDs, and comparing the number of relevant references to irrelevant ones in the models' answers.

To obtain the referenced answers in the zero-shot mode, we opted for the following prompt:

> Respond to the Instruction using only the information provided in the relevant abstracts under Abstracts. Reference the statements with the provided abstract_id in brackets next to the statement (for example PUBMED:1235):
> {instruction}

To obtain the referenced answers from the fine-tuned models, we use the following prompt:

> Respond to the Instruction using only the information provided in the relevant abstracts in “‘Abstracts“‘ below.
> {instruction}

Both prompts were chosen after extensive testing of several different prompting strategies and prompt versions.

We use default inference parameters for all four models, except setting the repetition_penalty to 1.1 for the fine-tuned models and varying the values of max_new_tokens (max_tokens for the zero-shot mode) for all four models. Despite trying to add the limit to the answers through the max_new_tokens parameter or through trying to add a limit to the prompt (e.g. "Answer in at most 300 words."), all the models continuously generated an arbitrary number of tokens. The same behavior was noticed in GPT-4 Turbo during the creation of the PQAref dataset. Token limitation, primarily imposed due to the prolonged inference time for higher values, often led to interrupted answers. Finally, the limit was set to 1225, to slightly exceed the longest complete answer length in the training dataset (see Section 3.2.1).

We refer to the zero-shot results of these two models as 0-M1 for v0.1 and 0-M2 for v0.2 and to the results of the fine-tuned models as M1 for v0.1 and M2 for v0.2. In both prompts, the *instruction* for the fine-tuned models consists of the user query and 10 retrieved abstracts. An example of a question and GPT-4 Turbo's answer from the test set, along with other four models' answers to the same question can be seen in Appendix A1.

Automated evaluation. The number of referenced abstracts in generated answers within PQAref test set (containing 908 examples) can be seen in Table 2. What can be observed is that 1 reference per answer is most common in GPT-4 Turbo answers from PQAref (241 answers). M1 and M2 have the highest number of answers with 3 references (185 cases for M1 and 178 for M2). In the case of zero-shot results, both 0-M1 and 0-M2 most commonly did not reference any abstracts in their responses: 527 occurrences (58% of all the answers) for 0-M1 and 165 for 0-M2 (18.2% of all the answers). M1 and M2 did not reference any abstracts in 8 (0.9%) and 5 (0.5%) answers, respectively. By manual inspection of these answers, the models stated that none of the abstracts were relevant, demonstrating their proficiency in task execution. On the other hand, in most of the answers without references 0-M1 and 0-M2 answered the question but without providing any references to their statements. Additionally, some 0-M2's answers (35 of them) repeated the first part of the instruction, suggesting the need for further postprocessing of its answers.

In the entire test set, comprising 908 examples with a total of 9080 abstracts (10 abstracts per example), 0-M2 has the highest average number of references per answer of 4.74, followed by M2 with 4.2 and M1 with 4.01, while 0-M1 produced 2.51 references per answer.

To measure the relevance of the referenced abstracts, we evaluated whether the models refer-

Table 2: Number of referenced abstracts per model on the PQAref test set. N: number of referenced abstracts per answer. TOTAL: is the sum of referenced abstracts per model. AVG: the average number of references per answer.

	Number of answers containing N references				
N	GPT-4 Turbo	0-M1	0-M2	M1	M2
0	2	**527**	**165**	8	5
1	**241**	27	11	86	105
2	76	66	47	138	112
3	128	28	92	**185**	**178**
4	126	17	114	172	169
5	119	25	110	117	124
6	87	28	94	72	75
7	45	26	61	66	34
8	29	47	64	27	34
9	31	47	83	22	23
10	24	70	67	15	49
TOTAL	3,464	2,285	4,307	3,648	3,816
AVG	3.81	2.51	4.74	4.01	4.20

enced at least the most relevant abstract for each question. Our dataset contains questions from PubMedQA, which in a number of cases originate from actual PubMed abstract titles. This means that during retrieval, the article whose title matches the question is very likely to be retrieved as relevant. In our test split, this indeed is the case in 823 out of 908 inputs. We decided to take such abstracts as the most relevant ones for those 823 inputs, which allowed us to automatically measure the number of times the models referenced that particular abstract. Table 3 presents the number of missed and referenced most relevant abstracts using this tactic. When looking at the GPT-4 Turbo answers from the test set, the most relevant article was missed in only one case, suggesting it served as a good referencing role model. M2 missed the relevant abstract in 10 examples, while M1 missed it in 29 examples. Overall, both fine-tuned models do reference the most relevant abstract in most cases (96.5% and 98.8% respectively). On the other hand, 0-M1 missed the most relevant abstracts in 60.4% answers and 0-M2 in 22.5% answers, which shows a significantly weaker ability of the models to identify and extract the most relevant abstracts compared to their fine-tuned versions.

We also evaluated whether all the IDs in the models' answers matched the PMIDs of context-provided abstracts to verify none of them were hallucinated. GPT-4 Turbo's answers in the PQAref dataset contained no hallucinated IDs. However, both M1 and M2 produced hallucinated IDs, with a notable discrepancy. M1 produced 79 hallucinated IDs, while M2 produced only 3. The hallucinated IDs differ from the actual IDs by one or two digits. Upon manual inspection of the answer content and referenced IDs, we found that M1 tended to blend information from various abstracts, whereas M2 utilized information solely from the corresponding abstract. This suggests that M2 exclusively hallucinated some of the digits from the existing abstract ID, but not the content. This behavior remains consistent across different temperature values of the model. Looking at the zero-shot performance, 0-M1 hallucinated 11 IDs. However, it also did not reference any abstracts in 58% of cases, which then presents an even higher number compared to the number of answers containing references. 0-M2 hallucinated in case of 26 IDs. The results point to a clear advantage of M2's answers in this respect.

Manual evaluation. To perform manual evaluation, we extracted 10 random examples from the PQAref test set. We then manually assessed the relevance of each of the abstracts in the examples. We generally distinguished between two types of abstracts: relevant and irrelevant. The abstracts we considered *relevant* were the ones that covered all the specific aspects of the question and thus provided direct answers. Among them, we defined the abstracts whose title matched the question as *the most relevant* (as mentioned for 823 examples during automatic evaluation). On the other hand, we identified two types of *irrelevant* abstracts. The first type includes abstracts that miss the main topic of the question (e.g. discuss heart failure instead of knee problems), which we considered *completely irrelevant* abstracts. The other type that discusses a more general topic and thus does not cover all the aspects of the question we considered *partially irrelevant*. This group could also be observed as the one that contains additional information but does not provide the direct answer to the question.

It is crucial to recognize that there can be two types of mistakes when irrelevant abstracts are concerned. If the model references a completely irrelevant abstract that is a clear mistake, however, if it references a partially irrelevant abstract, whether it is wrong may depend on the other references in the answer. If the answer also contains the reference that gives a direct answer to the question (relevant abstract), this could be considered additional information. If this is not the case, the model may have missed the main point.

Finally, we examined how the models referenced

Table 3: The number of missed and referenced most relevant abstracts of 823 abstracts across the models.

	GPT-4 Turbo	0-M1	0-M2	M1	M2
Relevant missed	1 (0.1%)	497 (60.4%)	185 (22.5%)	29 (3.5%)	10 (1.2%)
Relevant referenced	822 (99.9%)	326 (39.6%)	638 (77.5%)	794 (96.5%)	813 (98.8%)

the most relevant and irrelevant abstracts. For these 10 qualitatively observed examples, the fine-tuned models referenced the most relevant abstracts every time, meaning that they grasped the main point. On the other hand, 0-M1 and 0-M2 failed to reference these abstracts 4 and 2 times. Moreover, these answers of 0-M1 and 0-M2 contained no references whatsoever. None of the models referenced completely irrelevant abstracts. The general tendency of all four models was to provide additional information by referencing partially irrelevant abstracts. In several situations, the models seemed to filter the abstracts based on their understanding of a term used in the question, thus excluding the abstracts that use a different phrasing or an extended meaning of the term (e.g. donation taken to refer only to organ, tissue or bone marrow donation and not to cell and blood donation).

We also conducted a quantitative analysis to examine how well they identified all the relevant abstracts. To overcome variations in the number of relevant abstracts per document and document-specific characteristics, we considered all 100 abstracts, 10 for each of 10 questions, collectively.

Of these 100 abstracts, the evaluators identified 42 relevant and 58 irrelevant abstracts. We prioritized and calculated recall for relevant abstracts for each model, as our primary concern is their ability to correctly identify and reference relevant abstracts. M1 exhibited the highest recall of 0.76, followed by M2 with 0.67, 0-M2 with 0.62 and 0-M1 with 0.29. For reference, the recall measured on the GPT-4 Turbo answers from the test set totalled 0.62. These results are summed up in the first row of Table 4. The findings suggest that, based on the analysis of these 10 manually reviewed documents, M1 outperforms the other models in terms of referencing abstracts deemed relevant by evaluators, showing the highest benefit from the fine-tuning process.

4.3 System evaluation

In this section, we provide the preliminary joint evaluation of our system: the IR component (based on hybrid lexical and semantic search) and the generative component using the outputs of our IR,

Table 4: Recall values for relevant abstracts on 10 examples from the PQAref test set and same 10 questions with abstracts retrieved with our IR system.

	GPT-4 Turbo	0-M1	0-M2	M1	M2
PQAref	0.62	0.29	0.62	**0.76**	<u>0.67</u>
IR	0.46	0.37	<u>0.59</u>	**0.64**	0.58

We manually evaluated the IR output on the same 10 PQAref questions we chose for the evaluation of the generative component in Section 4.2. To retrieve the relevant abstract from indexed PubMed articles, we utilized the best-performing hybrid search parameter combination from Section 4.1 and retrieved 10 abstracts for each question. After manually determining the abstract relevance, we obtained 50% P@10. This metric underscores the effectiveness of our IR component in locating documents for query responses. The fact that IR evaluation on BioASQ reached the best performance of P@10 30.8% with the same combination of weights for hybrid search as manual evaluation on PQAref, further corroborates the results obtained in manual evaluation conducted on the PQAref dataset.

We then used the same prompt for GPT-4 Turbo as in Section 3.2.1, and the ones used in Section 4.2 for 0-M1, 0-M2, M1 and M2, to generate referenced answers based on the retrieved documents. We further computed the recall values for the relevant abstracts in the 10 generated answers and displayed them in the second row of Table 4. It is noticeable that, once again, the model that performed best is M1, with the recall of 0.64. This model cites a greater number of abstracts that contain the relevant answers compared to other models. Based solely on the recall, 0-M2 showed better results compared to M2, albeit by only 0.01. However, in one of 10 examples it did not provide any references to its elaborate answer. M2, as the third best model with recall of 0.58 properly referenced all the answers. From Table 2, we can also observe that the model with most references is 0-M2, but it also does not provide any references in 18.2% of the answers. Taking this important aspect into consideration, M2's answers prove more reliable compared to 0-M2. M2 shows a slightly lower recall compared to M1 because it has fewer references to abstracts that provide direct answers to the questions. Nonetheless, since the IR component consistently finds documents related to the topic, we give preference to M2's answers since they include more additional citations, offering more elaborate answers on the same topics. Here, GPT-4 Turbo had the recall of 0.46, while 0-M1 had the lowest recall of all the models (0.37), owing to a large number of answers with no references (5 out of 10).

5 Conclusions and future work

In this paper, we provide an overview of biomedical generative search with answers grounded in PubMed and referenced claims. Our aim was to develop a system capable of generating accurate and verifiable answers to biomedical questions while maintaining user sovereignty and leveraging open-source models.

Starting with our IR component, we discovered that employing a combination of lexical and semantic searches yields the highest precision score. Our system demonstrates an absolute improvement of 23.4% MAP@10 measure compared to the PubMed search engine. Through separate evaluations, we found that lexical search alone outperforms semantic search. However, integrating both approaches is advantageous for identifying instances lacking exact term matches, where semantic search contributes significantly. To enhance semantic search performance in IR, one future direction is to fine-tune these models on domain-specific data. This approach aims to improve the quality of embeddings in the biomedical domain, enabling them to encode domain-specific knowledge better, enhance contextual understanding, and ultimately improve IR performance.

Overall, the Mistral 7B Instruct models performed comparatively to GPT-4 Turbo in terms of the task of referenced QA. Based on the evaluation of the whole PQAref test set, M1 and M2 showed superior performance over 0-M1 and 0-M2 in referencing the most relevant abstracts, with M2 showing an improved performance of 2.3% over M1, 21.3% over 0-M2 and 59.2% over 0-M1. As a general trend, M2 includes more information in its answers.

All four models showed hallucinations when generating IDs of references. Once again, M2 performed best in this respect with only 3 mismatches in ID digits, followed by 0-M1 (11) and 0-M2 (26), with the worst performance of 79 hallucinated answers coming from M1. While M2 was still using correct information from the corresponding abstract, this point needs further attention. Exchanging the IDs with numerals (1-10) for each abstract during fine-tuning could potentially solve this issue. This is something we plan to try in the next iteration of the dataset and training.

In terms of recall values for relevant abstracts, based on the manual evaluation of 10 examples from PQAref test set both fine-tuned models performed better, exhibiting a 47% and 5% improvement over their versions in zero-shot mode. The situation is slightly different for the same 10 questions with abstracts retrieved using our IR. While recall values of M1 were still superior, especially compared to 0-M1 with 27% improvement, 0-M2 performed slightly better than M2 (1% difference). However, considering the number of hallucinations and answers with no references, we give clear preference to the fine-tuned models.

It is worth noting that on a small test set, we have achieved a comparable and sometimes even better performance than GPT-4 Turbo with much smaller, open-source and fine-tuned models. Having an automated evaluation method of referenced QA quality would allow us to make a more comprehensive comparison of our models to models such as GPT-4 Turbo. To supplement manual evaluation and speed up the evaluation process, (Gao et al., 2023) created a benchmark for automatic evaluation of citation quality and proposed an entailment model as a method for automatic quality check of references. We intend to further check the quality of answers on a larger sample using the same method. At this point, M2 gives more reliable answers, but the decision on which model to use in our final pipeline will be made after this evaluation.

Acknowledgments

This work is funded within the framework of the NGI Search project under grant agreement No 101069364. The resources for the fine-tuning process were provided to us by the National Platform for Artificial Intelligence of the Republic of Serbia.

References

Anakin.ai. 2024. Mistral 7b v0.2 base model, the new open source llm king is here. `https://anakin.ai/blog/mistral-7b-v0-2-base-model/`. Accessed: 2024-04-16.

BioASQ team. 2024. Bioasq12 challenge dataset. [Accessed 26 April 2024].

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. Chatbot arena: An open platform for evaluating llms by human preference. *arXiv preprint arXiv:2403.04132*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. *arXiv preprint arXiv:2305.14314*.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488, Singapore. Association for Computational Linguistics.

Eugene Garfield. 1970. Can citation indexing be automated? *Essays of an Information Scientist*, 1:84–90.

Jocelyn Gravel, Madeleine D'Amours-Gravel, and Esli Osmanlliu. 2023. Learning to fake it: Limited responses and fabricated references provided by chatgpt for medical questions. *Mayo Clinic Proceedings: Digital Health*, 1(3):226–234.

Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently Teaching an Effective Dense Retriever with Balanced Topic Aware Sampling. In *Proceedings of SIGIR 2021*.

Jie Huang and Kevin Chen-Chuan Chang. 2024. Citation: A key to building responsible and accountable large language models. *arXiv preprint arXiv:2307.02185*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Nelson Liu, Tianyi Zhang, and Percy Liang. 2023. Evaluating verifiability in generative search engines. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7001–7025, Singapore. Association for Computational Linguistics.

Yu A Malkov and Dmitry A Yashunin. 2018. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE transactions on pattern analysis and machine intelligence*, 42(4):824–836.

Jacob Menick, Maja Trebacz, Vladimir Mikulik, John Aslanides, Francis Song, Martin Chadwick, Mia Glaese, Susannah Young, Lucy Campbell-Gillingham, Geoffrey Irving, and Nat McAleese. 2022. Teaching language models to support answers with verified quotes. *arXiv preprint arXiv:2203.11147*.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. 2022. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*.

Nan Yang, Tao Ge, Liang Wang, Binxing Jiao, Daxin Jiang, Linjun Yang, Rangan Majumder, and Furu Wei. 2023. Inference with reference: Lossless acceleration of large language models. *arXiv preprint arXiv:2403.12077*.

Shen Zheng, Jie Huang, and Kevin Chen-Chuan Chang. 2023. Why does ChatGPT fall short in providing truthful answers? *arXiv preprint arXiv:2304.10513*.

Appendix A: Example answers of generative LLMs

Table A1: Models' answers to the question "Does leisure time physical activity in early pregnancy protect against pre-eclampsia?"

GPT-4 Turbo
The relationship between leisure time physical activity (LTPA) in early pregnancy and the risk of pre-eclampsia is not clearly established, with studies showing mixed results. Some studies suggest that there may be a protective effect of LTPA against pre-eclampsia, while others do not find a significant association or suggest an increased risk with high levels of activity. A prospective cohort study in Denmark found that moderate levels of physical activity (1-270 minutes/week) in the first trimester had no statistically significant association with the risk of pre-eclampsia, but high levels of activity (exceeding 270 minutes/week) were associated with an increased risk of severe pre-eclampsia (PUBMED:19055653). Similarly, a population-based prospective cohort study in Amsterdam reported that the amount of time or intensity of physical activity in leisure time was not associated with a difference in risk of pre-eclampsia or gestational hypertension (PUBMED:19916877). In contrast, a study from the New Haven, Connecticut area suggested that women who engaged in any regular LTPA during early pregnancy were at decreased risk of pre-eclampsia (PUBMED:15466498). Another study found a tendency towards a lower risk of pre-eclampsia in women with the highest degree of physical activity during leisure time, especially in overweight women, although no significant associations were found (PUBMED:20121498). A systematic review examining the association between LTPA before and/or during pregnancy and the risk of pre-eclampsia found mixed results, with some studies suggesting that vigorous-intensity LTPA may reduce the risk of pre-eclampsia, while one study reported an elevated risk with high amounts of LTPA (PUBMED:23836014). Another case-control study found that women who performed regular LTPA during the first 20 weeks of pregnancy had a reduced risk of pre-eclampsia and gestational hypertension (PUBMED:2592903). In summary, the evidence is inconclusive regarding the protective effect of LTPA in early pregnancy against pre-eclampsia. While some studies suggest a potential benefit, especially with moderate activity levels, others indicate no effect or an increased risk with high levels of activity. It is important to consider the intensity and duration of LTPA when evaluating its impact on pre-eclampsia risk.
0-M1
The instruction is to provide a response to the instruction using only the information provided in the abstracts. The instruction is: "Does leisure time physical activity in early pregnancy protect against pre-eclampsia?" Based on the provided abstracts, the answer is: No, leisure time physical activity in early pregnancy does not protect against pre-eclampsia.
0-M2
The available evidence from the provided abstracts suggests that the relationship between leisure time physical activity (LTPA) in early pregnancy and the risk of pre-eclampsia is not clear-cut. Some studies indicate a protective effect of LTPA against pre-eclampsia, while others do not find a significant association or suggest an increased risk with high levels of activity. One prospective cohort study in Denmark found that moderate levels of physical activity (1-270 minutes/week) in the first trimester had no statistically significant association with the risk of pre-eclampsia, but high levels of activity (exceeding 270 minutes/week) were associated with an increased risk of severe pre-eclampsia (PUBMED:19055653). Another population-based prospective cohort study in Amsterdam reported that the amount of time or intensity of physical activity in leisure time was not associated with a difference in risk of pre-eclampsia or gestational hypertension (PUBMED:19916877). However, a study from the New Haven, Connecticut area suggested that women who engaged in any regular LTPA during early pregnancy were at decreased risk of pre-eclampsia (PUBMED:15466498). Another study found a tendency towards a lower risk of pre-eclampsia in women with the highest degree of physical activity during leisure time, especially in overweight women, although no significant associations were found (PUBMED:20121498). A systematic review examining the association between LTPA before and/or during pregnancy and the risk of pre-eclampsia found mixed results, with some studies suggesting that vigorous-intensity LTPA may reduce the risk of pre-eclampsia, while one study reported an elevated risk with high amounts of LTPA (PUBMED:23836014). Another case-control study found that women who performed regular LTPA during the first 20 weeks of pregnancy had a reduced risk of pre-eclampsia and gestational hypertension (PUBMED:2592903). In summary, the evidence is inconclusive regarding the protective effect of LTPA in early pregnancy against pre-eclampsia. While some studies suggest a potential benefit, especially with moderate activity levels, others indicate no effect or an increased risk with high levels of activity. It is important to consider the intensity and duration of LTPA when evaluating its impact on pre-eclampsia risk. Further research is needed to clarify the relationship between LTPA and pre-eclampsia risk.

M1

Based on the provided abstracts, leisure time physical activity in early pregnancy does not appear to protect against pre-eclampsia. A prospective cohort study in Denmark found that the two highest levels of physical activity were associated with increased risk of severe pre-eclampsia compared with the nonexercising group, suggesting that leisure time physical activity exceeding 270 minutes/week in first trimester may increase the risk of severe pre-eclampsia (PUBMED:19055653).

Another study in Amsterdam, The Netherlands, did not find any association between physical activity in leisure time early in pregnancy and the incidence of preeclampsia or gestational hypertension (PUBMED:19916877). Similarly, a study in New Haven, Connecticut, found that women who engaged in any regular leisure time physical activity, regardless of caloric expenditure, were at decreased risk of preeclampsia, but there was no indication of a protective effect of workplace activity, leisure-time physical activity, or unemployment (PUBMED:15466498). These findings suggest that while certain forms of physical activity may have a protective effect on preeclampsia, the overall evidence does not support a clear protective effect of leisure time physical activity in early pregnancy against this condition. Therefore, based on the available abstracts, leisure time physical activity in early pregnancy does not protect against pre-eclampsia.

M2

Based on the provided abstracts, there is conflicting evidence regarding whether leisure time physical activity (LTPA) in early pregnancy protects against pre-eclampsia. Some studies suggest that LTPA may offer protection against pre-eclampsia, while others do not find a significant association or even indicate that excessive LTPA could increase the risk of severe pre-eclampsia. For instance, one study found that women who engaged in any regular LTPA regardless of caloric expenditure were at decreased risk of preeclampsia, although another study did not find a significant association between the amount of time or intensity of physical activity in leisure time and the risk of preeclampsia or gestational hypertension (PUBMED:15466498; PUBMED:19916877). Another study suggested that women who performed regular LTPA had a reduced risk of pre-eclampsia and gestational hypertension, with the relative risks decreasing as the average time spent in LTPA increased (PUBMED:2592903). However, another study documented an increased risk of severe pre-eclampsia with higher levels of LTPA (PUBMED:19055653). Similarly, a systematic review reported that high-intensity LTPA before and/or during pregnancy or more than 4 hours per week of LTPA may reduce the risk of pre-eclampsia, but also noted that one study found an elevated risk of severe pre-eclampsia with high amounts of LTPA (PUBMED:23836014).

Furthermore, a case-control study indicated that women who performed regular LTPA had a reduced risk of pre-eclampsia and gestational hypertension, with the relative risks decreasing as the average time spent in LTPA increased (PUBMED:26910608). A systematic review and meta-analysis of RCTs and cohort studies supported the promotion of LTPA in pregnancy as a strategy to improve maternal and child health, suggesting that participation in LTPA was associated with lower weight gain during pregnancy, lower likelihood of gestational diabetes mellitus (GDM), and lower risk of preterm delivery (PUBMED:27282925).

In conclusion, while some studies suggest that LTPA in early pregnancy may protect against pre-eclampsia, particularly at moderate intensities, the evidence is not consistent across all studies. Further research is needed to clarify the relationship between LTPA and pre-eclampsia risk, taking into account the intensity and duration of physical activity, as well as individual differences in susceptibility to the condition. Additionally, the potential influence of other factors such as prepregnancy BMI and socioeconomic status should be considered (PUBMED:32093248).

Low-resource ICD Coding of Hospital Discharge Summaries

Ashton Williamson[1], **David de Hilster**[2], **Amnon Meyers**[3],
Nina Hubig[1], **and Amy Apon**[1]

[1]School of Computing, Clemson University
[2]LexisNexis Risk Solutions
[3]Conceptual Systems LLC
taw2@clemson.edu

Abstract

Medical coding is the process by which standardized medical codes are assigned to patient health records. This is a complex and challenging task that typically requires an expert human coder to review health records and assign codes from a classification system based on a standard set of rules. Since health records typically consist of a large proportion of free-text documents, this problem has traditionally been approached as a natural language processing (NLP) task. While machine learning-based methods have seen recent popularity on this task, they tend to struggle with codes that are assigned less frequently, for which little or no training data exists. In this work we utilize the open-source NLP programming language, NLP++, to design and build an automated system to assign International Classification of Diseases (ICD) codes to discharge summaries that functions in the absence of labeled training data. We evaluate our system using the MIMIC-III dataset and find that for codes with little training data, our approach achieves competitive performance compared to state-of-the-art machine learning approaches.

1 Introduction

Medical coding is the process by which healthcare institutions assign standardized codes to patient health records for downstream use in applications such as statistical analysis, indexing patient health records, coding medical billing claims (Moriyama et al., 2011), and assessing quality of patient care (O'Malley et al., 2005). While these systems of classification represent a critical infrastructure with extensive significance in the healthcare domain, the success of their implementation largely rests on the efficient and precise assignment of these codes to a patient's health record. We focus on the task of assigning International Classification of Diseases (ICD) codes to patient health records, which is a complex, multi-stage process with many possible points of failure, often resulting in improperly assigned or missing codes. The Department of Health and Human services found in 2010 that approximately half of all claims for evaluation and management services were incorrectly coded, resulting in $6.7 billion in improper payments by Medicare (Levinson et al., 2014). Medical coding thus presents itself as a critical task which would greatly benefit from increased automation.

Since much of the information required for assigning codes is contained within unstructured text documents, the problem of medical coding has traditionally been approached through the framework of natural-language processing (NLP). Much research has been conducted in this area but it remains a challenging problem. Inherent limitations of state-of-the-art approaches tend to restrict their practical utility (Dong et al., 2022). Challenges include large label spaces and long document lengths, user requirements for explainability, and adaptability to local facility needs and medical advances. Classical machine-learning and deep-learning based approaches tend to be limited by the need for quality labeled data for supervised training. Due to restrictions on the distribution of patient medical data, collecting and curating useful datasets for these tasks is a major challenge (Johnson et al., 2016a; Searle et al., 2020; Johnson et al., 2016b). Annotation costs to develop gold-standard datasets can be prohibitive (Searle et al., 2020).

In this paper we propose a system for ICD coding that provides support for explainability and functions without training data. Our system first extracts medical entities from an input document, then maps these entities to concepts in the Unified Medical Language System (UMLS) (Bodenreider, 2004). Finally, we use these concepts as terms to assign ranking scores to ICD-9 codes for the input note. We use the openly available MIMIC-III dataset to evaluate the performance of our system and to compare to existing state-of-the-art ap-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 548–558
August 16, 2024. ©2024 Association for Computational Linguistics

proaches to the task. Our contributions are to:

- Design and implement an automated end-to-end scalable system using readily available medical knowledge sources to assign ICD-9 codes to discharge summaries,
- Compare our approach to state-of-the-art deep learning approaches, and
- Demonstrate the utility of knowledge-based algorithms for low-resource ICD coding

2 Background

2.1 International Classification of Diseases

The ICD is a standardized nomenclature and classification system for diseases and medical procedures, which was originally intended to facilitate the statistical analysis of health data (Moriyama et al., 2011). Each successive revision to the ICD, typically spanning 10-20 years, has sought to address new use cases while adapting to advances in medicine and healthcare, and has continued to grow in number of total codes. The tenth version, ICD-10, has nearly 72,000 procedure codes. We utilize the ICD-9-CM, a clinical modification of the ICD-9 adapted for use in the US, which contains well over 10,000 codes. Each entity within the ICD-9-CM is encoded by a unique identification string consisting of three to five digits and an optional single letter prefix corresponding to a supplementary category (see Figure 1). Practical applications of the ICD in healthcare have expanded and now have come to include the indexing of health record data in hospitals, the coding of medical billing claims (Moriyama et al., 2011), and the assessment of quality of patient care (O'Malley et al., 2005).

2.2 MIMIC

To evaluate our approach to ICD coding we use the Medical Information Mart for Intensive Care (MIMIC) dataset (Johnson et al., 2016b). MIMIC is an openly accessible database of de-identified electronic health record data for patients admitted to the intensive care unit of the Beth Israel Deaconess Medical Center. There are four releases of the MIMIC dataset. Our work focuses on the third release as full access to the fourth release was not available until late in the project. The third release, MIMIC-III, was published in 2016 and contains data for 53,423 distinct hospital admissions. Each hospital admission record is comprehensive and includes data in one of the following categories: billing, descriptive, dictionary, interventions, laboratory, medications, notes, physiologic, and reports. MIMIC-III includes free text notes and reports such as radiology reports and hospital discharge summaries as well as ICD-9 codes for a hospital admission, making it a useful resource for the development and evaluation of automated code extraction.

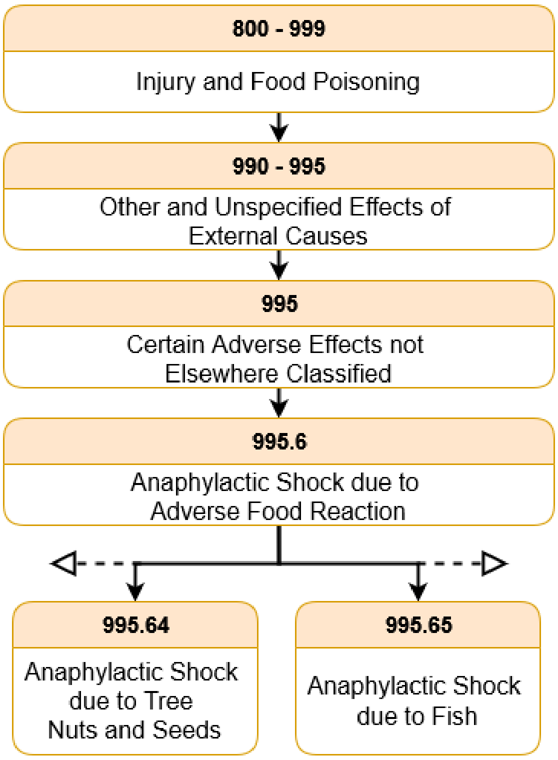

800 - 999
Injury and Food Poisoning
990 - 995
Other and Unspecified Effects of External Causes
995
Certain Adverse Effects not Elsewhere Classified
995.6
Anaphylactic Shock due to Adverse Food Reaction
995.64
Anaphylactic Shock due to Tree Nuts and Seeds
995.65
Anaphylactic Shock due to Fish

Figure 1: Example path in the ICD-9 hierarchy.

2.3 Unified Medical Language System

The UMLS is a repository of biomedical vocabularies and associated tools that is developed and maintained by the US National Library of Medicine. The UMLS consists of the Metathesaurus, a biomedical thesaurus that links concepts from different constituent vocabularies; the Semantic Network, which defines semantic types and provides relationships between UMLS concepts; and the SPECIALIST Lexicon, an English dictionary that includes biomedical terms (US National Library of Medicine, 2009). The Metathesaurus is a collection of source vocabularies including biomedical thesauri, classification systems, coding systems, and controlled term lists such as SNOMED-CT (Stearns et al., 2001). Terms in these vocabularies are linked to standard identifiers using semantic or lexical information. The Semantic Network defines 127 different semantic types for concepts as well as 54 different relationships between them, comprising a network in which types are the nodes and relationships are the vertices. We leverage this concept structure along with the relationships defined in the Semantic Network to map key terms found in the text to concepts in the UMLS.

The Specialist Lexicon is a dictionary containing both common English terms as well as domain-specific medical terms that was developed with the express purpose of aiding in natural language processing of medical text (US National Library of Medicine, 2009). The lexicon includes key linguistic information for each term, including spelling variations, conjugations or conjugation patterns, plural forms, and more (Browne et al., 2000). The NLM releases a set of utilities for working with the specialist lexicon. Of these utilities, we use the Lexical Variant Generator to generate variants and synonyms of terms within the ICD-9 code titles.

2.4 NLP++

We select the natural language processing programming language NLP++ to implement our system (Deane et al., 2001). NLP++ utilizes a *multi-pass, multi-strategy* architecture in which each user-defined pass over the input text performs a specific step in processing or parsing the text. Passes are broken down into specific regions: rule regions, code regions and declarative regions. Rule regions perform operations on the parse tree using predefined operators, code regions include code which is executed at runtime, and declarative regions include user-defined functions that can be called from both rule and code regions. The multi-pass strategy constructs a single, best-first parse tree which is refined by each successive pass. Sequences of passes are grouped together as an *analyzer* tailored to a particular application.

NLP++ also incorporates a hierarchical knowledge base management system that allows the programmer to dynamically store and use information extracted from input texts. The conceptual grammar includes both knowledge bases and dictionaries. Knowledge bases allow the user to store and retrieve hierarchically structured information while dictionaries consist of entries and corresponding key/value pairs. After tokenization, a lookup is performed on the parse tree. Nodes that match dictionary entries are tagged with their respective key/value pairs. This facility constitutes a key aspect of building effective analyzers for parsing text.

3 Related Work

Though research on automated medical coding dates as least as far back as the 1970's (Powsner, 1978; Stanfill et al., 2010), access to data and hardware limitations prevented the development of large-scale solutions. The first work on ICD coding was published in the 1990s (Larkey and Croft, 1995). It treated the task as one of information retrieval, employing k-nearest-neighbors, relevance feedback, and Bayesian classifiers to select and rank relevant codes. Other early approaches leveraged biomedical entity recognition systems to extract clinically-significant entities which could then be linked to codes from the target coding system (Barrows Jr et al., 2000; Friedman et al., 2004). While these approaches saw some success on test datasets, they were limited by their ability to generalize to new datasets and their ability to scale to larger label spaces.

Medori and Fairon examined the automated assignment of ICD-9-CM codes to French language clinical notes (Medori and Fairon, 2010). Their system was bipartite, including an extraction step using both dictionary-based and heuristic methods to identify relevant coding information and a classification step using Naïve Bayes classifiers to assign codes. Classifiers were built for codes that appeared more than five times in the corpus, resulting in only 1,497 classifiers. The approach separates the task into an extraction and classification step and inspires our approach to isolating relevant context which is then used for code classification.

Mullenbach et al. (Mullenbach et al., 2018) implement an attentional convolutional network to assign ICD-9 codes to discharge summaries in the MIMIC-III dataset. They introduce train, development and test splits for the Full set of MIMIC-III discharge summaries as well as a Top 50 split that includes only the 50 most frequently assigned codes. Both the Full and Top-50 splits defined by Mullenbach et al. have become the standard for comparison in the literature (Yang et al., 2022). We evaluate our system on the test set of these splits.

Yang et al. (Yang et al., 2022) address the long-tail challenge of ICD coding by both defining a rare code subset of the MIMIC-III dataset and introducing a training algorithm to improve performance on rare codes. The rare disease subset, MIMIC-III-

	Full	**Top 50**	**Rare 50**
Number of Notes	52,723	11,368	391
Number of Patients	41,126	10,356	386
Number of Unique Codes	8,922	50	50
Mean Codes per Note	15.9	5.7	1.0
Train/Dev/Test %	91/3/6	71/14/15	61/5/34

Table 1: Splits of the MIMIC-III dataset, including Full (Mullenbach et al., 2018), Top 50 (Mullenbach et al., 2018), and Rare 50 (Yang et al., 2022).

rare50, includes less-common codes in MIMIC-III and corresponding discharge summaries. The motivation for this subset comes from the observation that 4,115 of the 8,692 unique codes in the MIMIC-III dataset occur fewer than 6 times (Yang et al., 2022). To create this set, the authors first select codes with fewer than 10 occurrences, then select the top 50 from this set after splitting between train and test. Common diseases are also manually removed, resulting in 50 codes in total. Given the reliance of pretrained language models on labeled data, few- and zero-shot settings like those introduced in the Rare-50 subset pose a challenging problem. We use the Rare-50 split to evaluate our system in a low-resource setting.

4 Methods

We extract all notes from the NOTEEVENTS table in the MIMIC-III dataset (version 1.4) with CATEGORY field matching "Discharge Summary", including both reports and addenda. Notes where the ISERROR flag is set are dropped and all addendum-type discharge summaries are concatenated to their corresponding original reports, following previous work (Mullenbach et al., 2018). ICD9 codes are then generated from the PROCEDURES_ICD and DIAGNOSES_ICD tables using the subject and hospital admission IDs.

4.1 Domain Knowledge Integration

The UMLS serves two key roles in our system. The first is to identify clinically significant terms within ICD-9 titles. The second is to resolve ambiguous domain-specific language. To the first end, we utilize the UMLS term mapping utility to normalize terms within ICD-9 titles by mapping them to alphanumeric lexical identifiers in the Specialist Lexicon, known as Entry Unique Identifiers (EUIs). These terms can be single words or n-grams within the title. For example, ICD-9 code 285.1 with title *Acute posthemorrhagic anemia*, generates EUIs E0007202, *acute posthemmorhagic anemia*; E0049207, *posthemmorhagic*; E0007127, *acute*; and E0008920, *anemia* (Figure 2, Step 1).

In the second step, we leverage the normalized terms identified in the first step to generate sets of alternative forms of these terms (see Step 2 of Figure 2). These alternative forms, or *variants*, include at minimum abbreviations, acronyms, plural forms, conjugations, and spelling variations. To accomplish this, we use the Lexical Variant Generation (LVG) command line utility included with the Specialist Lexicon tools (Sherertz et al., 1989). The LVG takes a term or list or terms as input and outputs a list of variants according to the specified flow control options. These options include normalization methods like stripping punctuation and diacritics, and splitting ligatures as well as derivational options like generating fruitful variants, inflections, synonyms, and spelling variants. We utilize the fruitful variants flag, which includes spelling variants, inflections, synonyms, acronyms and abbreviations, expansions of abbreviations and acronyms, and derivations (Divita et al., 2014). This information is aggregated and organized into knowledge bases and dictionaries for downstream use in our analyzer.

4.2 Note Processing

In the note processing stage, we take a set of notes, in this case each of the test sets of MIMIC-III, and output a set ICD-9 codes for each note. Our approach involves three steps: extraction, linking and ranking. In the first step we decompose the input text into structured sections. In the second step we extract key terms and link these to central concepts. In the third step we rank the set of extracted codes using an inverse-document frequency-based method. In this section we give an overview of the analyzer structure and experimental setup to inves-

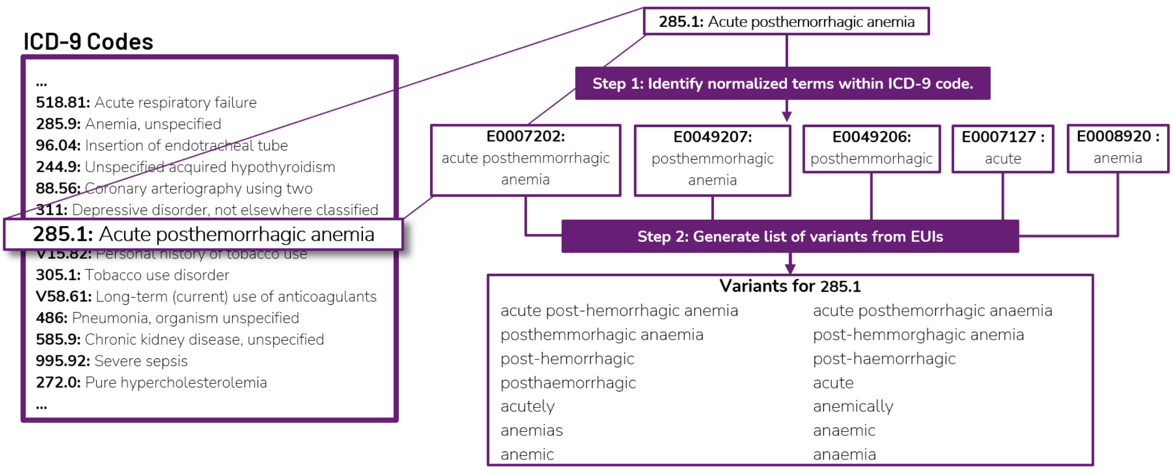

ICD-9 Codes
...
518.81: Acute respiratory failure
285.9: Anemia, unspecified
96.04: Insertion of endotracheal tube
244.9: Unspecified acquired hypothyroidism
88.56: Coronary arteriography using two
311: Depressive disorder, not elsewhere classified
285.1: Acute posthemorrhagic anemia
V15.82: Personal history of tobacco use
305.1: Tobacco use disorder
V58.61: Long-term (current) use of anticoagulants
486: Pneumonia, organism unspecified
585.9: Chronic kidney disease, unspecified
995.92: Severe sepsis
272.0: Pure hypercholesterolemia
...
285.1: Acute posthemorrhagic anemia
Step 1: Identify normalized terms within ICD-9 code.
E0007202: acute posthemmorrhagic anemia
E0049207: posthemmorhagic anemia
E0049206: posthemmorhagic
E0007127 : acute
E0008920 : anemia
Step 2: Generate list of variants from EUIs
Variants for 285.1
acute post-hemorrhagic anemia
acute posthemorrhagic anaemia
posthemmorhagic anaemia
post-hemmorghagic anemia
post-hemorrhagic
post-haemorrhagic
posthaemorrhagic
acute
acutely
anemically
anemias
anaemic
anemic
anaemia

Figure 2: Outline of the pipeline for term normalization and variant generation for ICD-9 titles.

tigate the effects of knowledge sources and ranking formulations on overall ICD coding performance.

We first describe the general structure of the NLP++ analyzer. Our analyzer consists of 27 distinct passes (for a comprehensive list with pass types, see Appendix A) each of which performs a distinct step in note processing. The first step for NLP++ analyzers is the tokenization step in which we perform tokenization of the input note using the built-in *Dictionary Tokenizer* in NLP++. The Dictionary Tokenizer uses a word-based tokenization strategy which splits the text on whitespace and punctuation. Additionally, strings containing both digits and letters are split into tokens containing either all letters or all numbers. The tokenization pass also performs lookups in the dictionaries for each token in the parse tree. If a token matches an entry in one of the dictionaries, its attributes are added to the corresponding parse tree node. In the case that a multi-token phrase is matched, the entire token sequence match is reduced to a *_phrase* node in the parse tree. The output of the tokenization step is a shallow parse tree consisting of tokens from the input text which are tagged with negation type and an integer EUI identifier, where applicable.

The next three passes of the analyzer (i.e., *KBFuncs*, *array_funcs*, and *pn_funcs*) are declarative passes that define functions which are called throughout the analyzer. *KBFuncs* is a library pass that provides useful functions for working with knowledge bases. We utilize NLP++ built-in functions to add unique strings, concepts, and values to knowledge bases along with functions which facilitate exporting knowledge bases. In the *array_funcs* pass we define functions to perform common array operations, including array concatenation, element swapping, QuickSort, binary search, duplicate filtering, and conversion functions for interoperability with knowledge base data structures. The *pn_funcs* pass includes a single function that appends a value to a parse tree node's variable.

Passes 6 through 19 perform cleaning of the note text and organization of the parse tree. Starting with pass 6 we excise non-relevant information including de-identified placeholder strings, headers and footers, which empirical investigation suggests predominantly contain metadata. After filtering, we organize the parse tree into structural components in order of increasing granularity: sections, subsections, enumerated lists, and sentences. For sections and subsections, header names (e.g., "History of Patient Illness" or "Chief Complaint") are added as attributes on the parent node when present. Finally, we clean all whitespace from the parse tree, including space characters, tabs, and newlines.

Pass 20, *gather_negations*, implements the NegEx algorithm with a maximum distance of 5 nodes between a negation term and a clinical entity. Our negation window size follows the original NegEx implementation (Chapman et al., 2001) for its relative effectiveness and ease of implementation, though some work has shown improvement using a dynamic window size (Meystre and Haug, 2005). The first rule in the pass matches a leaf node tagged as *pre*-negation, along with the next 5 sibling nodes or up to the next *_section* /*_subsection* /*_sentence* boundary, whichever comes first. Since compound medical terms are reduced to a single *_phrase* node, they are treated as a single entity, or node match. The next rule performs the same operation for *post*-negation terms, instead excising the preceding 5 nodes.

Passes 21-27 perform the term extraction and ranking steps. The aim of these steps is to take the structured parse tree with terms tagged for normalization and rank the importance of the terms using a term-frequency inverse-document-frequency based method (Sparck Jones, 1972). The set of all ICD-9 titles is the reference corpus, *C*. We start by copying all tagged terms in the parse tree onto parent nodes, so that each *section*, *subsection*, and *sentence* node contains a list of all normalized terms, represented by unique identifiers, contained within.

For any given ICD-9 code $c \in C$, we represent it as a set of terms that occur within its title such that $T_c = \{t_1, t_2, ..., t_n\}$ (for an example, see Figure 2). We then calculate the frequency of each unique term within all ICD-9 titles, given by f_t, to encode the relative specificity of each term (lower corpus frequency => higher specificity) (Sparck Jones, 1972). We then define the total weight of a code, $\mathcal{W}_c$, as the sum of the inverse document frequencies (IDFs) of each of its constituent terms, $t \in T_c$ over the ICD-9 corpus, C, or $\mathcal{W}_c = \sum_i^{|T_c|} \frac{f_{t_i}}{|C|}$. We then use the same IDF term weights to calculate the ranking score of a code with respect to a particular note. Let $G = t_1, t_2, ..., t_m$ be the set of all terms in the document of interest and $H = G \cap T_c$ be the set of codes in both the input text and the code c, then the rank of code c with respect to the note, $\mathcal{R}_c$, is given by the following:

$$\mathcal{R}_c = \frac{\sum_{j=1}^{|H|} \mathcal{W}_{H_j}}{\mathcal{W}_c} \tag{1}$$

By dividing by the total possible code weight, we ensure that the ranking score for a code is not dependent on the number of terms within its title. Note that unlike TF-IDF, we are not taking into account the frequency of a term within the note.

Since we are ranking a code based on the occurrence of its constituent terms within the target text, we hypothesize that constraining term matches to smaller sections of the text will lead to better performance. To test this we re-formulate our ranking function by assigning a weight to each code for each section s and aggregate the ranking score for each code by applying an aggregation function: *max*, *mean*, or *sum*. We experiment with ranking codes at the *section* and *sentence* level.

4.3 Evaluation

We evaluate all approaches on the test sets of the Top 50 and Rare 50 splits of MIMIC-III (for a comparison of these splits, see Table 1). Following previous work (Mullenbach et al., 2018; Yang et al., 2022), we use the receiver operating characteristic area under the curve (*ROC AUC*), F1-score and precision at k, for $k = 5$. Since ICD-coding is a multi-class classification problem, we provide ROC AUC and F1-score results using both *macro*- and *micro*-averaging.

4.4 Execution Characteristics

The note processing stage is conducted in parallel on the Clemson Palmetto HPC cluster using only CPUs. Notes are first written to individual text files, which are then mapped to available processes with GNU Parallel (Tange, 2022). Each process runs an instance of the note processing analyzer in the NLP++ engine. The final pass in the analyzer, Pass 27, writes the analyzer results to a single-line CSV file containing the hospital admission ID (HADM ID) for the discharge summary followed by ranking scores for all ICD-9 codes, in predetermined order. Each of these output files is read and appended to a single CSV file which is indexed by HADM ID and has columns corresponding to ranking scores for each ICD code. This final step is also performed in parallel using GNU Parallel to coordinate the process.

5 Results

We denote each of our methods as follows: *LexSyn* refers to the use of lexical variants and synonyms for normalization. The subscript refers to the aggregation method-*max*, *sum*, or *mean*-and the scope of term matches-*sent* for sentence-level, *sect* for section level, and *full* for the full note.

5.1 Rare 50

Results for the Rare 50 split are shown in Table 3. We find that our LexSyn-Section$_{\text{max}}$ analyzer achieves a level of performance comparable to recent state-of-the-art approaches in terms of ROC AUC, falling within 4 points of KEPTLongformer$_{\textit{finetuned}}$, the best-performing deep-learning model.

Despite the slight performance difference, we identify a few key advantages which support the

	ROC AUC		F1-Score		Prec. @ k
Approach	Micro	Macro	Micro	Macro	5
CAML (Mullenbach et al., 2018)	91.1	87.5	52.4	60.6	61.1
PLM-ICD (Huang et al., 2022)	91.9	89.3	67.6	64.3	61.7
MSMN (Yuan et al., 2022)	94.7	92.8	72.2	68.1	67.6
KEPTLongformer (Yang et al., 2022)	94.2	92.0	72.7	68.5	67.4
LexSyn-Section$_{\text{max}}$	69.8	70.9	33.3	37.1	29.4
LexSyn-Section$_{\text{mean}}$	67.8	68.9	30.1	33.7	26.3
LexSyn-Section$_{\text{sum}}$	68.6	71.3	31.7	39.6	27.6
LexSyn-Sent$_{\text{max}}$	69.6	69.1	31.3	34.4	29.0
LexSyn-Sent$_{\text{mean}}$	**71.7**	72.2	**34.5**	37.1	**31.8**
LexSyn-Sent$_{\text{sum}}$	70.0	**72.2**	32.9	**40.8**	27.7
LexSyn-Full	68.0	68.0	31.9	38.5	29.6

Table 2: Results on the MIMIC-III Top 50 test set (Mullenbach et al., 2018). Results for all approaches are run to completion. The best performing result for each metric from our approaches is bolded.

	ROC AUC		F1-Score	
Approach	Micro	Macro	Micro	Macro
MSMN$_{\text{pretrained}}$	76.2	75.3	17.1	17.2
MSMN$_{\text{zero-shot}}$	48.9	52.3	3.5	4.0
MSMN$_{\text{finetuned}}$	44.0	58.2	3.3	4.2
KEPTLongformer$_{\text{p}}$	82.3	81.4	30.9	25.8
KEPTLongformer$_{\text{z}}$	76.5	74.9	16.7	15.2
KEPTLongformer$_{\text{f}}$	83.3	82.7	32.6	30.4
LexSyn-Section$_{\text{max}}$	**77.8**	80.0	**12.6**	24.7
LexSyn-Section$_{\text{mean}}$	76.7	77.2	12.4	23.4
LexSyn-Section$_{\text{sum}}$	77.0	80.4	12.5	20.8
LexSyn-Sent$_{\text{max}}$	76.2	**81.0**	10.2	28.2
LexSyn-Sent$_{\text{mean}}$	74.0	77.6	8.8	28.2
LexSyn-Sent$_{\text{sum}}$	75.9	80.8	10.3	**29.2**
LexSyn-Full	**77.8**	80.0	12.2	23.6

Table 3: **MIMIC-III Rare 50 test set results**. Results for previous approaches from (Yang et al., 2022). The best performing result for each metric from our approaches is bolded.

utility of our analyzer in a clinical setting. The first of these is the potential for explainability, as described in the next section. Our system is fully traceable and provides evidence from the text to support a particular code assignment. Furthermore, our approach does not require any training data (labeled or unlabeled) which is advantageous in a low-resource setting.

5.2 Top 50

Results for the Top 50 split are shown in Table 2. Results on the Top 50 split are comparable to the results for the Rare 50 split but not as close to recent state-of-the-art deep learning methods on the Top 50 split. We note that the Top 50 split has a smaller label space (50 labels vs 8,922 for the full set) and a large number of samples per label, making this dataset significantly less challenging for deep learning methods than the Rare 50 and Full sets. We nonetheless find that our approaches achieve a reasonable performance baseline.

We perform additional analysis on individual codes to explain the resulting code predictions. Observation of the per-code F1-scores is shown in Figure 3. Performance for individual codes on the Top 50 dataset is highly variable, with F1 scores that range from nearly 0.9 to 0.0. We conduct analysis for the three ICD-9 codes with individual F1 scores equal to 0.0 (*412*, *285.1*, and *39.61*). We plot the confusion matrices, seen in Figure 4, for these codes and observe for these codes a positive label is never or almost never predicted.

For code *412*, "Old myocardial infarction", manual inspection reveals that this code is almost exclusively assigned when "myocardial infarction", or its initialism MI, occurs in the *Past Medical History* section of the discharge summary. In fact, applying a simple matching rule for these terms in the *Past Medical History* section significantly outperforms our approach, with an F1 score on the Top 50 test set of 58.2. Although our system does not leverage code title information to restrict code matches to sections in the text, our system does provide support for incorporating this type of rule.

For code *285.1*, "Acute posthemorrhagic anemia", we find that the terms themselves do not appear in the text. We suspect that the indicator of this code comes from blood sample results, for which

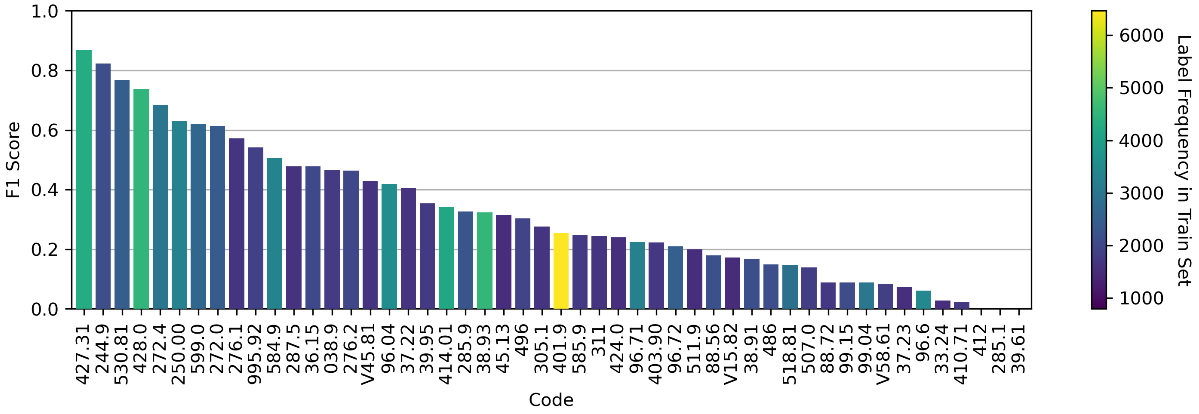


Figure 3: F1-score per code on the Top-50 dataset, sorted by decreasing F1-score. Bar colors represent frequency of the code in the Top-50 training set.

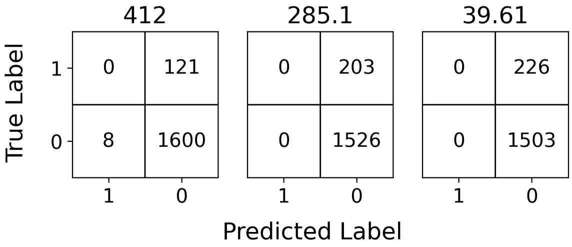


Figure 4: Confusion matrices for the three codes in the Top 50 set with F1-Score equal to 0.0.

abnormal red blood cell counts and hemoglobin levels are marked by an asterisk. This type of inference from non-textual signifiers or numerical data is outside the current scope of our analyzer, though one could add a heuristic rule to help identify these cases.

For code *39.61*, “Extracorporeal circulation auxiliary to open heart surgery”, we find that the title and key subphrases of the title do not occur as such in the text. In the set of discharge summaries selected for review, we observe the presence of procedures which may classify as extracorporeal circulation methods, for example “CPB”, an initialism for cardiopulmonary bypass. Further investigation reveals that cardiopulmonary bypass (UMLS CUI: C0007202) is defined as a *narrower*, or child, concept of extracorporeal circulation (UMLS CUI: C0015354). This suggests that leveraging ontological information beyond just synonyms may be helpful for improving performance.

5.3 Codes with Multiple Occurrences

For the Top 50 and Rare 50 datasets, our analyzer generates code ranks for each sentence or section. When the same code occurs in multiple sentences or sections the result is a large number of ranking scores for a code in the note. To handle the situation in which a code occurs multiple times in the same note, we use one of three aggregation methods: the mean, the sum or the median of the ranks. However, we suspect that noisy ranking scores for frequent terms adversely affect performance.

6 Future Work

Our approach to extracting clinical entities does not differentiate between semantic interpretations of a particular medical entity. This is particularly salient for abbreviations and acronyms, which often require contextual clues to disambiguate (Savova et al., 2008). Consider, for example, the term ‘ms’, which maps to 12 unique concepts in the 2007AC UMLS (Savova et al., 2008). Our system would, in practice, give equal weight to each of these *senses* of the term ‘ms’ without attempting to identify the true sense of the term in the text. An lucrative path for future work may be to incorporate heuristic algorithms for word-sense disambiguation (WSD) (Schuemie et al., 2005; Chasin et al., 2014) into the entity extraction passes of the analyzer.

In our system we utilize the Lexical Variant Generator of the UMLS to identify variants of key medical terms. This allows us to normalize these variants in the text by mapping them to a central concept. We experiment with different variant generation setups as outlined in Section 4.1. We find the literature on lexical normalization for medical entities to be sparse (Divita et al., 2014; Hedberg, 2013). An in-depth analysis of the downstream impact of different variant generation setups would be a useful tool for guiding the construction of systems that utilize the lexical tools. Additionally, we hypothesize that our system could better leverage existing ontological information, including free text descriptions and hierarchical relationships.

The dataset on which we evaluate our system, MIMIC-III (Johnson et al., 2016b), is a large and meticulously compiled dataset which has realized significant progress toward the systematic study of medical coding. Due to the increasing ubiquity of MIMIC in the literature on medical coding, the validation of ground truth labels in the dataset has become supremely important (Searle et al., 2020). As our system is developed and built entirely from available medical sources using knowledge-based algorithms, the system output is consistent and reproducible. It is potentially usable as a companion to deep learning based methods to aid in the development of gold-standard labels for the MIMIC dataset.

Perhaps the most critical area for future research that we identify is the potential of our approach to be utilized as an assistive software for human coders. In this role our system does not replace human coders but can be used as a “first pass” to coding or to flag inconsistencies for human verification. Code for our system is made available at `https://github.com/ashtonomy/low_resource_icd_coding`.

Limitations

Despite showing competitive performance in few- or zero-shot settings, our analyzer is limited by its performance in high-resource settings, such as the Top 50 test set discussed in Section 5. More work is needed to improve performance in this domain before deployment in a clinical setting is considered. We also note that our system is evaluated on medical notes sourced from a single hospital system. In general, we find that more robust evaluation on data from different source domains is needed to more effectively gauge performance. As discussed in 6, this is a challenge at present due to limited access to openly available annotated medical notes.

Ethics Statement

In this natural language processing research we adhere to the highest ethical standards to ensure integrity, transparency, and respect for all contributors and subjects involved. We recognize the potential impacts of NLP technologies on society and are committed to responsible research practices. This version of the system is a proof-of-concept and is intended for research purposes only. It has not been validated for production use.

Acknowledgments

The authors acknowledge the generous support of HPCC Systems and LexisNexisRisk. This material is based on work supported by the National Science Foundation under Grant Nos. MRI# 2024205, MRI# 1725573, and CRI# 2010270. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Clemson University is acknowledged for their generous allotment of compute time on the Palmetto Cluster.

References

Randolph C Barrows Jr, M Busuioc, and Carol Friedman. 2000. Limited parsing of notational text visit notes: ad-hoc vs. nlp approaches. In *Proceedings of the AMIA Symposium*, page 51. American Medical Informatics Association.

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic Acids Research*, 32(suppl_1):D267–D270.

Allen C Browne, Alexa T McCray, and Suresh Srinivasan. 2000. The specialist lexicon. *National Library of Medicine Technical Reports*, pages 18–21.

Wendy W. Chapman, Will Bridewell, Paul Hanbury, Gregory F. Cooper, and Bruce G. Buchanan. 2001. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of Biomedical Informatics*, 34(5):301–310.

Rachel Chasin, Anna Rumshisky, Ozlem Uzuner, and Peter Szolovits. 2014. Word sense disambiguation in the clinical domain: a comparison of knowledge-rich and knowledge-poor unsupervised methods. *Journal of the American Medical Informatics Association*, 21(5):842–849.

Paul Deane, David de Hilster, and Amnon Meyers. 2001. Text processing in an integrated development environment (ide): Integrating natural language procesing (nlp) techniques. *PC AI*, 15(5):36–40.

Guy Divita, Qing T Zeng, Adi V Gundlapalli, Scott Duvall, Jonathan Nebeker, and Matthew H Samore. 2014. Sophia: a expedient umls concept extraction annotator. In *AMIA Annual Symposium Proceedings*, volume 2014, page 467. American Medical Informatics Association.

Hang Dong, Matúš Falis, William Whiteley, Beatrice Alex, Joshua Matterson, Shaoxiong Ji, Jiaoyan Chen, and Honghan Wu. 2022. Automated clinical coding: what, why, and where we are? *NPJ digital medicine*, 5(1):159.

Carol Friedman, Lyudmila Shagina, Yves Lussier, and George Hripcsak. 2004. Automated Encoding of Clinical Documents Based on Natural Language Processing. *Journal of the American Medical Informatics Association*, 11(5):392–402.

RoseMary Hedberg. 2013. Analyzing lexical tool's fruitful variants for concept mapping in the synonym mapping tool. *NLM Associate Fellowship Projects, nlm.nih.gov*.

Chao-Wei Huang, Shang-Chi Tsai, and Yun-Nung Chen. 2022. PLM-ICD: Automatic ICD coding with pretrained language models. In *Proceedings of the 4th Clinical Natural Language Processing Workshop*, pages 10–20, Seattle, WA. Association for Computational Linguistics.

Alistair E. W. Johnson, Mohammad M. Ghassemi, Shamim Nemati, Katherine E. Niehaus, David A. Clifton, and Gari D. Clifford. 2016a. Machine learning and decision support in critical care. *Proceedings of the IEEE*, 104(2):444–466.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016b. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

Leah S Larkey and W Bruce Croft. 1995. Automatic assignment of icd9 codes to discharge summaries. Technical report, Technical report, University of Massachusetts at Amherst, Amherst, MA.

Daniel R Levinson, D Grant, J Durley, R Bessette, and M Verges. 2014. Improper payments for evaluation and management services cost medicare billions in 2010. *Department of Health and Human Services*.

Julia Medori and Cedrick Fairon. 2010. Machine learning and features selection for semi-automatic icd-9-cm encoding. In *Louhi '10: Proceedings of the NAACL HLT 2010 Second Louhi Workshop on Text and Data Mining of Health Documents*. Association for Computational Linguistics.

Stéphane M Meystre and Peter J Haug. 2005. Comparing natural language processing tools to extract medical problems from narrative text. In *AMIA annual symposium proceedings*, volume 2005, page 525. American Medical Informatics Association.

Iwao Milton Moriyama, Ruth M Loy, Alastair Hamish Tearloch Robb-Smith, Harry Michael Rosenberg, and Donna L Hoyert. 2011. History of the statistical classification of diseases and causes of death. *DHHS publication*, 2011-1125.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. 2018. Explainable prediction of medical codes from clinical text. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1101–1111, New Orleans, Louisiana. Association for Computational Linguistics.

Kimberly J O'Malley, Karon F Cook, Matt D Price, Kimberly Raiford Wildes, John F Hurdle, and Carol M Ashton. 2005. Measuring diagnoses: Icd code accuracy. *Health services research*, 40(5p2):1620–1639.

Seth M Powsner. 1978. Automatic coding of medical problem lists. *Yale Medicine Thesis Digital Library*, (3041).

Guergana K Savova, Anni R Coden, Igor L Sominsky, Rie Johnson, Philip V Ogren, Piet C De Groen, and Christopher G Chute. 2008. Word sense disambiguation across two domains: biomedical literature and clinical notes. *Journal of biomedical informatics*, 41(6):1088–1100.

Martijn J Schuemie, Jan A Kors, and Barend Mons. 2005. Word sense disambiguation in the biomedical domain: an overview. *Journal of Computational Biology*, 12(5):554–565.

Thomas Searle, Zina Ibrahim, and Richard Dobson. 2020. Experimental evaluation and development of a silver-standard for the MIMIC-III clinical coding dataset. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, pages 76–85, Online. Association for Computational Linguistics.

DD Sherertz, MS Tuttle, NE Olson, MS Erlbaum, and SJ Nelson. 1989. Lexical mapping in the umls metathesaurus. In *Proceedings of the Annual Symposium on Computer Application in Medical Care*, page 494. American Medical Informatics Association.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

Mary H Stanfill, Margaret Williams, Susan H Fenton, Robert A Jenders, and William R Hersh. 2010. A systematic literature review of automated clinical coding and classification systems. *Journal of the American Medical Informatics Association*, 17(6):646–651.

Michael Q Stearns, Colin Price, Kent A Spackman, and Amy Y Wang. 2001. Snomed clinical terms: overview of the development process and project status. In *Proceedings of the AMIA Symposium*, page 662. American Medical Informatics Association.

Ole Tange. 2022. Gnu parallel 20220722 ('roe vs wade'). GNU Parallel is a general parallelizer to run multiple serial command line programs in parallel without changing them.

US National Library of Medicine. 2009. Umls reference manual. Https://www.ncbi.nlm.nih.gov/books/NBK9676/.

Zhichao Yang, Shufan Wang, Bhanu Pratap Singh Rawat, Avijit Mitra, and Hong Yu. 2022. Knowledge injected prompt based fine-tuning for multi-label few-shot icd coding. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing. Conference on Empirical Methods in Natural Language Processing*, volume 2022, page 1767. NIH Public Access.

Zheng Yuan, Chuanqi Tan, and Songfang Huang. 2022. Code synonyms do matter: Multiple synonyms matching network for automatic icd coding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 808–814.

A Analyzer Pass Structure

#	Pass	Type
1	dicttokz	Tokenizer
2	KBFuncs	@DECL
3	array_funcs	@DECL
4	pn_funcs	@DECL
5	init_kb	@CODE
6	clean_notes	@RULES
7	get_negation	@RULES
8	get_breaks	@RULES
9	get_sections	@RULES
10	get_loose_passages	@RULES
11	group_loose_passages	@RULES
12	remove_breaks	@RULES
13	get_subsection_headers	@RULES
14	get_subsections	@RULES
15	get_list_items^{R}	@RULES
16	get_lists	@RULES
17	get_sentences	@RULES
18	sentences	@RULES
19	remove_whitespace	@RULES
20	gather_negations	@RULES
21	shift_keywords	@RULES
22	keyword_funcs	@DECL
23	set_line_count	@CODE
24	extract_codes	@RULES
25	rank_codes	@CODE
26	aggregate_and_predict	@CODE
27	kb_out	@CODE

Table 4: Pass structure in the NLP++ ICD-coding analyzer for MIMIC-III notes. Passes marked ^{R} are recursive.

Towards ML-supported Triage Prediction in Real-World Emergency Room Scenarios

Faraz Maschhur[1] **Klaus Netter**[2] **Sven Schmeier**[1] **Katrin Ostermann**[2]
Rimantas Palunis[3] **Tobias Strapatsas**[3,4] **Roland Roller**[1]
[1]German Research Center for Artificial Intelligence (DFKI)
[2]DNC Information Management GmbH
[3]Städtische Kliniken Mönchengladbach
[4]Klinik für Akut- und Notfallmedizin Asklepios Klinikum Harburg

Abstract

In emergency wards, patients are prioritized by clinical staff according to the urgency of their medical condition. This can be achieved by categorizing patients into different labels of urgency ranging from immediate to not urgent. However, in order to train machine learning models offering support in this regard, there is more than approaching this as a multi-class problem. This work explores the challenges and obstacles of automatic triage using anonymized real-world multi-modal ambulance data in Germany.

1 Introduction

The differentiation of treatment urgency is an important step in clinical emergency medicine. Various validated triage systems have been established for this purpose, and in Germany, their use is virtually mandatory. In practice, this means that within the first 10 minutes of a patient's arrival, assigning a treatment priority and thus setting a time target until contact with a medical professional is a required process.

According to the Manchester Triage System (MTS), the possible triage level ranges from immediate to non-urgent, which is mainly meant as guidance to lead employees in the emergency rooms (ER) in making their triage decisions. Although only five triage levels exist, the problem is not as straightforward as it seems. The triage model of MTS follows a decision tree, where on the first level, a so-called diagram or lead symptom (diagnosis) is determined, and on the second level, indications (discriminator) specific to the selected diagram are identified. The indications translate to predefined triage levels, where the most urgent triage level among them expresses the severity of the case. Since the diagrams and indications are not defined by sharp boundaries, it may well happen that a medical professional reaches the same indication through different diagrams. So, the same triage level can be decided on by choosing different, but equally valid indications.

The data used in this work is a mixture of multiple text fields describing the situation of the patient and some first diagnosis (in the form of text), and a large set of structured information, i.e. medical measurement of vital signs, age or sex. In particular, vital signs such as temperature, oxygen saturation, etc. are an essential part of the MTS model defining an indication.

In this work, we have built prototypical machine learning models using retrospective data for automatic triage in the emergency ward and examine the results and obstacles of our approaches. More specifically, we examine to which extent a transformer-based BERT model can address the problem of noisy, unbalanced, semi-structured multi-class real-world data. Different training strategies are explored, particularly to deal with the different interconnected classes as well as to deal with the varying label frequencies. Moreover, we investigate how we can extend a given BERT model, which is normally only suitable for text data, by additional structured information. Finally, we test an approach to build up a hybrid model, combining machine learning with a rule-based component.

2 Related Work

Various studies so far have looked at the possibilities of automatic triage but differ in terms of data, models/solutions, target, and results. Stewart et al. (2023) provide an overview of different triage use cases strongly related to NLP. However, many approaches target, for instance, text (Bergman et al., 2023) or a mix of structured and unstructured (text) data (Klug et al., 2020; Arnaud et al., 2023) to predict a binary label, such as mortality or hospitalization. Some others focus on a larger number of

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 559–569
August 16, 2024. ©2024 Association for Computational Linguistics

triage labels used in emergency care, such as Levin et al. (2018); Sarbay et al. (2023). Depending on the given data, solutions such as gradient-boosting (Klug et al., 2020) or BERT-based approaches (Arnaud et al., 2023), also in a hybrid setup (Wang et al., 2023), are popular. Also, with the rise of large language models, LLM-based solutions have been tested (Frosolini et al., 2024; Levine et al., 2023; Sarbay et al., 2023). So far, however, there are no studies that have automatically determined treatment priorities based on data from emergency services, nor are there any that predict such a large number of different classes simultaneously as we are trying to do.

In this work, we deal with different types of data - (partly sequential) numerical, categorical, and text data. To handle such data types, many different approaches and architectures exist, to combine different 'modalities', such as for instance mapping all different information into one vector space (Rethmeier et al., 2020), the combination of transformers with linear layers and LSTMs (Yang and Wu, 2021; Deznabi et al., 2021), or LLMs with time series (Jin et al., 2023). However, in this work, we rely on a simple architecture based on transformers, integrating different data types and exploring how far we can get.

3 Data and Methods

This work is based on anonymized ambulance reports with triage assignments from a German emergency ward covering two years and including more than 18k cases. The data was recorded electronically and contains a wide variety of different information—overall, about 600 different features exist, ranging from binary to numeric and sequential (e.g., sequences of particular measurements during the ride in the ambulance). The data includes information such as age, sex, blood pressure, pain score, information about consciousness, burns, medications, or motoric skills. In addition to the structured information, the data also includes text fields, describing the emergency situations, an initial diagnosis, injuries, symptoms, as well as the original cause of alarm. An example of a patient case is provided in the Appendix.

The data represents real-world data and has been labeled with the triage categories, consisting of a diagram (diagnosis) and an indication (discriminator), by the emergency department staff in accordance with the MTS. As mentioned, the selected diagram limits the possible indications, and each indication directly translates to a triage level. Several different diagrams and indications may be valid, but only one of each is annotated—in the case of indications, it is the most urgent one. Even if several equally serious indications may apply, only one is labeled. According to MTS, 54 diagrams, 125 indications and 5 triage levels exist. However, due to the real-world context of the data, some labels do not occur in the dataset at all. Figure 1 provides an overview of the label distribution of indications and diagrams in the data. A more detailed overview is provided in the Appendix.

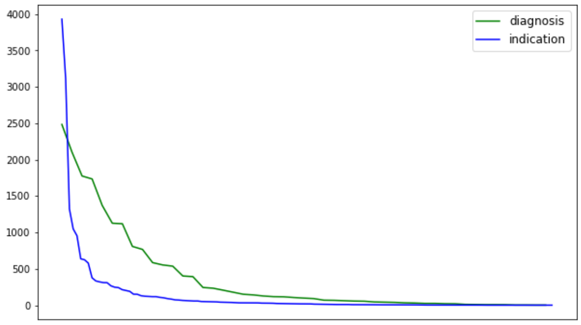


Figure 1: Distribution of diagrams and indications with the two most frequent diagrams *Discomfort in adults* (2480) and *Falls* (2104) and the two most frequent indications *Recent problem* (3931) and *Moderate pain* (3116). For more details see Table 13 and Table 14.

3.1 Data Challenges

Due to the nature of the data properties and the collection procedure, the dataset used in this work poses non-trivial challenges. Since the data has been gathered in the real-world, there is some amount of noise incorporated into the data. The text fields have been filled in by many different paramedics and the abbreviations are not standardized. In a few cases, patients' symptoms resolved between data collection and arrival at the hospital, resulting in a different label than suggested by the data collected. Additionally, the distribution of each of the three labels is unbalanced, with diagnosis and indication having many distinct labels resulting in a long tail problem, as shown in Figure 1 and Tables 13 & 14.

Some of the diagram or indication categories are similar to each other in how they are assessed but differently impact the triage process. Extensive experience guide medical professionals in choosing between these categories. For example, the two indications *Abnormal cardiac history* and *Cardiac* *pain*, often only differ in how the medical professional assesses the state of the patient, while being associated with different triage levels.

Moreover, in many cases, the text features are not sufficient for the successful prediction of diagrams or indications. Non-text features like temperature, oxygen saturation and others can be crucial to identify certain diagrams or indications. For instance, the *Very Hot* indication is given at a body temperature above 41°C or *Hyperglycemia* is defined as a glucose level above 17 mmol/l. This limits the model's ability to learn from text features alone since these values are not necessarily included in the text features. Since only a single diagram and a single indication per data point were labeled, although several may apply, the data is less effective in training, as correlations between the diagram or indication classes are not learnable.

3.2 Models

For all our experiments, we rely on medBERT.de (Bressem et al., 2023) and examine different setups, with respect to how we train the model, as well as the input we consider. In Training we first differentiate between using the standard cross-entropy loss (normal), to establish a basic baseline, versus weighted cross-entropy loss (weighted). In addition, we examine models trained independently on each class (single) versus multi-task models (MT) trained on all three classes at the same time. In the MT setup, each class is trained together with the other target classes, and during training the focus (in terms of loss) slowly shifts towards the target class, as depicted in Figure 2.

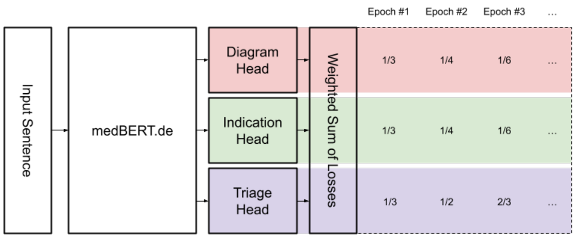


Figure 2: Training of a multi-task model with a focus on triage class - with each epoch the loss contribution optimizes towards triage class

In addition, we test different setups regarding input data: as stated before, not all relevant information for correct classification is given through text data. We therefore examine how additional structured information (pain score, temperature, blood sugar level, heart rate, diastolic/syst. blood pressure, age, sex) could be inserted into the BERT-based solution. In the first setup, we translate structured data into a single sentence using expert knowledge and add these sentences to the standard text data using [SEP] tokens. We refer to this approach as 'extra as text'. For instance, the pain score is translated into a sentence such as '*Pat. has [no/slight/moderate/severe/very severe/the worst imaginable] pain.*'[1] The mapping of numeric values into categories is done by medical guidelines.

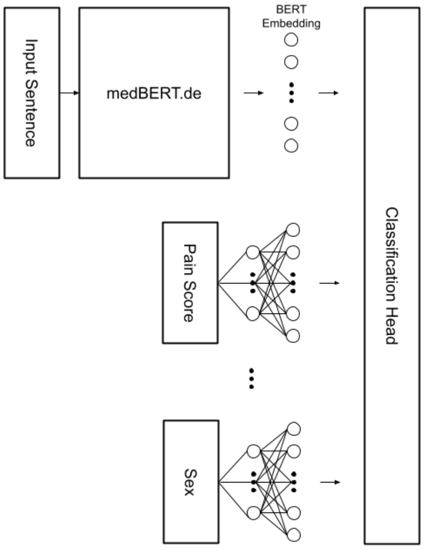


Figure 3: Overview of the architecture in the extra as feature approach - each extra feature is scaled-up through a two-layer MLP and is then inserted, together with the output of medBERT.de, into a classification head

Alternatively, in the second setup, we scale the features through two-layer MLPs and process them in a custom classification head together with the BERT embeddings of the standard text data, as depicted in Figure 3. We refer to this approach as 'extra as feature'. An advantage of this approach is that no bias is introduced through the manual translation into sentences. Since many labels do not occur frequently, the integration of a rule-based component that processes structured information seems helpful in certain scenarios. For this, we examine if an external, rule-based component using expert knowledge targeting vitals could be easily integrated into our system. We refer to this data as 'expert'. This data is also integrated into our model through the use of [SEP] tokens. Every model,

[1] As we work with German we use this translated pattern: '*Pat. hat [keine/leichte/mäßige/starke/sehr starke/stärkste vorstellbare] Schmerzen.*'. More examples can be found in the Appendix.

except the one using the standard cross-entropy loss (normal), incorporates class weights to address the dataset's unbalanced label distribution.

4 Experiments

4.1 Setup

For our experiment, we randomly split the data into training, development, and test sets (80/10/10%). Patient cases that were missing labels were removed, as well as labels that occurred only once. All models have been trained with early stopping and then applied to the test data and evaluated using precision, recall, and F1 (weighted & macro).

4.2 Results

Table 1 presents the weighted and macro F1 scores of different single-class and multi-task models. As the data contains a large number of different classes, which are unbalanced, it is not surprising that macro scores are generally much lower than weighted scores, particularly for indications, which include more than 120 labels. In the single model setup it is difficult to see any additional value of training the BERT model with weighted loss. What we can see, however, is that additional information (extra (text/feat) and expert) seems to have a positive impact on the model performance. In many cases, the impact is particularly visible in the case of macro F1. Most notable here is the inclusion of the simple, expert model.

Table 1: Performance according to F1 weighted (w) and macro (m) of (upper part) single models and (middle and lower part) multi-task models on triage data, including the prediction of the triage label (P) and the deduction of the triage label from the predicted indication (D).

	Diagram		Discrimin.		Triage (P)		Triage (D)	
	w	m	w	m	w	m	w	m
normal	0.592	0.384	0.33	0.102	0.54	0.34	0.542	0.334
weighted	0.607	0.401	0.272	0.12	0.539	0.356	0.528	0.33
extra (text)	0.607	0.414	0.279	0.132	0.542	0.349	0.533	0.338
extra (feat.)	0.587	0.415	0.232	0.133	0.536	0.325	0.513	0.305
expert	0.608	0.416	0.303	0.147	0.55	0.362	0.545	**0.379**
MT weighted	0.588	0.377	0.325	0.145	0.55	0.363	**0.564**	0.354
MT extra-text	0.6	0.411	0.323	0.136	**0.576**	0.379	0.549	0.368
MT extra-feat	**0.613**	0.41	0.316	0.113	0.558	0.392	0.544	0.314
MT expert	0.6	0.415	**0.331**	0.152	0.574	**0.415**	0.554	0.367
MT exp.&ext.-text	0.612	**0.428**	0.328	**0.157**	0.575	0.403	0.552	0.35
MT exp.&ext.-feat	0.599	0.393	0.27	0.121	0.556	0.389	0.548	0.375

Comparing the single and multi-task models, the table shows a clear tendency that multi-task models perform better than the single models. Again, this improvement can be particularly seen in the macro F1 evaluation. More notable (only included in the Appendix), is that our multi-task learning leads to improvements for the given target class. The multi-task models that combine the different expert and extra features generally appear to provide the best approach, especially the model including extra-text.

Table 1 depicts two approaches to predict the triage level: *Triage (P)* represents the direct prediction of triage labels and *Triage (D)* represents the deduction of the triage level from the predicted indication label. In the emergency ward, *Triage (D)* would be the regular way how to solve the problem. In many cases *Triage (P)* provides slightly better results, in terms of weighted and macro F1, compared to the deduction. However, while the direct approach sees triage labels as uncorrelated classes, in reality they are correlated. It certainly makes a difference, given a gold label *red* (immediate), if we predict *orange* (very urgent) or *green* (standard), as orange is closer to red and also more urgent. For this reason, we calculate the MRSE (mean root squared error) using the model *MT expert & extra-text* and for *Triage (P)* achieve a value of 0.588, and for *Triage (D)* a score of 0.525. This indicates that the deduction might be the better choice, as the deduction provides labels closer to the gold label.

Table 2: Top-3 performance according to F1 weighted (w) and macro (m) of a selection of models.

	Diagram		Discrimin.	
	w	m	w	m
normal	**0.86**	0.607	0.572	0.247
MT weighted	0.737	0.517	**0.633**	0.285
MT expert	0.859	0.612	0.628	**0.293**
MT exp. & ext.-text	0.843	0.589	0.624	0.28
MT exp. & ext.-feat	0.836	**0.631**	0.581	0.287

One of the challenges handling this data is that multiple diagram and indication labels can be valid, but only one is annotated. This can have an influence on the performance of our models in case we predict valid labels different to the annotation in the dataset. In order to examine this we evaluate our models by considering the top-3 predictions of diagrams and indications, as depicted in Table 2. The results show, in all cases, a very strong boost in performance, particularly for diagrams. In the case of indication, the weighted score achieves 0.633, while the macro score still remains below 0.3, which might be due to the long tail problem and the fact that many indications require additional structured information.

4.3 Analysis & Discussion

As the data includes a large variety of labels with a long tail problem - and many of the cases occur only a few times - the task is very challenging. At the same time, many labels do not depend solely on the text features. Therefore, it is difficult to detect them unless the included text contains a clear hint. For instance, for the indication *Suspected Sepsis*, a patient needs to have at least two of the following symptoms: new onset of confusion, increased respiratory rate (above 22/min) or low blood pressure (below 100 mmHg systolic), where the last two symptoms depend on structured data. While our top-3 approach tries to overcome the multiple labels problem, the moderate results for top-3 indications show the limitations of pure text-based approaches for the triage classification problem. We assume that more structured data needs to be included in the model to better deal with labels that are less connected to text data. Moreover, it might be beneficial to include additional rule-based components/predictions, in order to deal with the long tail problem. Data-driven machine learning is popular, but if data is sparse or expensive to gather rule-based components might be a valid approach to overcome its problems.

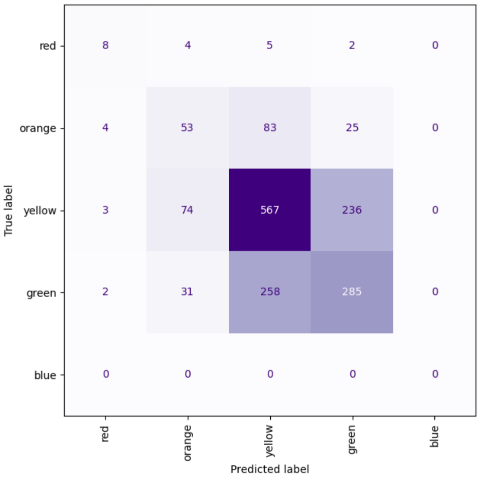


Figure 4: Confusion matrix of top-1 Triage (D) label prediction from red to blue (very urgent to not urgent).

Instead of tackling the problem with a pure text-based transformer approach, we achieve better results by integrating additional data. Text-based integration appears to be more promising than the feature-based approach. Unfortunately, the text-based approach is not scalable, as we need to deal with the model's limited input size. Therefore, the feature-based approach combining BERT embeddings with additional features might be the best approach for using more structured features.

In addition to the missing labels and the existing noise, many labels are generally difficult to predict because they are very abstract, such as *Recent Problem*. According to the definition 'A problem that occurred within the last week is referred to as a recent problem'. Although very general, it is, still one of the most frequent labels in our data, and similar others exist.

While unbalanced data is a problem for machine learning, in a real-world setup for triage prediction, it makes a difference if a patient is accidentally predicted with a triage label that is too high or too low. At the same time, particularly the very urgent classes are most important to predict correctly. Figure 4 depicts the confusion matrix for the top-1 triage label prediction. The figure shows, for instance, that various cases are assigned with a higher triage label and a similar number of cases with a lower triage label, which could risk a patient's life. Even more seriously, various of the patients labeled as red (immediate treatment) are labeled with a lower label. In order to introduce a (hybrid) machine learning system for automatic triage, this is the most important problem to address. Figure 5 (Appendix) shows an alternative confusion matrix when we apply the top-3 indication prediction, infer the triage level, and always choose the most urgent one. This might be a possibility to reduce triage predictions below the gold label. However this approach still offers space for improvements.

5 Conclusion

In this work, we presented a challenging real-world problem to support employees in an emergency ward. Although the data is multi-modal (numerical and text), we approached the problem with text-based transformer solutions. Considering the difficulties with noise, missing labels, the number of different labels, and the long tail problem, the results are promising. However, we foresee that we need to include additional information as extra features to further boost the performance and to provide models with a more substantial benefit in an emergency ward.

Limitations

The presented solution still has many limitations, as presented in the discussion. Naturally noise has some impact on the model's performance, but overall, we also need to investigate how to boost the performance further and particularly examine how we perform in really urgent cases. While misclassification is negligible for uncritical cases, it is certainly not in very critical ones.

Ethical Statement

Experiments have been conducted on retrospective data. Therefore, our model does not directly impact patient treatments. In the foreseen application, the model is intended to be integrated into an assistance and decision-support system (when good enough), providing additional information for the human performing the actual triage. Where possible, the medical personnel will be provided with explanations and further details corroborating the suggested categorizations.

The project is based on a comprehensive protocol to ensure privacy and data protection. For the model's training and testing, the retrospective data has been completely anonymized and stripped of any personal, local, and temporal information that would allow reference to patients or medical personnel involved.

Acknowledgments

The project has received funding from the German Federal Ministry of Education and Research (BMBF) through the project KIBATIN (16SV9040).

Special thanks to Michael Deppe and Florian Oetke from DNC, who implemented the ICE-Analysis platform and application into which the categorizer is integrated and who were essential in providing us with the training and test material as well as valuable feedback to our approach.

References

Emilien Arnaud, Mahmoud Elbattah, Pedro A Moreno-Sánchez, Gilles Dequen, and Daniel Aiham Ghazali. 2023. Explainable nlp model for predicting patient admissions at emergency department using triage notes. In *2023 IEEE International Conference on Big Data (BigData)*, pages 4843–4847. IEEE.

Erik Bergman, Luise Dürlich, Veronica Arthurson, Anders Sundström, Maria Larsson, Shamima Bhuiyan, Andreas Jakobsson, and Gabriel Westman. 2023. Bert based natural language processing for triage of adverse drug reaction reports shows close to human-level performance. *PLOS Digital Health*, 2(12):e0000409.

Keno K. Bressem, Jens-Michalis Papaioannou, Paul Grundmann, Florian Borchert, Lisa C. Adams, Leonhard Liu, Felix Busch, Lina Xu, Jan P. Loyen, Stefan M. Niehues, Moritz Augustin, Lennart Grosser, Marcus R. Makowski, Hugo JWL. Aerts, and Alexander Löser. 2023. Medbert.de: A comprehensive german bert model for the medical domain. *arXiv preprint arXiv:2303.08179*.

Iman Deznabi, Mohit Iyyer, and Madalina Fiterau. 2021. Predicting in-hospital mortality by combining clinical notes with time-series data. In *Findings of the association for computational linguistics: ACL-IJCNLP 2021*, pages 4026–4031.

Andrea Frosolini, Lisa Catarzi, Simone Benedetti, Linda Latini, Glauco Chisci, Leonardo Franz, Paolo Gennaro, and Guido Gabriele. 2024. The role of large language models (llms) in providing triage for maxillofacial trauma cases: a preliminary study. *Diagnostics*, 14(8):839.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. 2023. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*.

Maximiliano Klug, Yiftach Barash, Sigalit Bechler, Yehezkel S Resheff, Talia Tron, Avi Ironi, Shelly Soffer, Eyal Zimlichman, and Eyal Klang. 2020. A gradient boosting machine learning model for predicting early mortality in the emergency department triage: devising a nine-point triage score. *Journal of general internal medicine*, 35:220–227.

Scott Levin, Matthew Toerper, Eric Hamrock, Jeremiah S Hinson, Sean Barnes, Heather Gardner, Andrea Dugas, Bob Linton, Tom Kirsch, and Gabor Kelen. 2018. Machine-learning-based electronic triage more accurately differentiates patients with respect to clinical outcomes compared with the emergency severity index. *Annals of emergency medicine*, 71(5):565–574.

David M Levine, Rudraksh Tuwani, Benjamin Kompa, Amita Varma, Samuel G Finlayson, Ateev Mehrotra, and Andrew Beam. 2023. The diagnostic and triage accuracy of the gpt-3 artificial intelligence model. medrxiv. *Published online February*, 1(2023):10–1101.

Nils Rethmeier, Necip Oguz Serbetci, Sebastian Möller, and Roland Roller. 2020. Efficare: better prognostic models via resource-efficient health embeddings. In *AMIA Annual Symposium Proceedings*, volume 2020, page 1060. American Medical Informatics Association.

İbrahim Sarbay, Göksu Bozdereli Berikol, and İbrahim Ulaş Özturan. 2023. Performance of emergency triage prediction of an open access natural language processing based chatbot application (chatgpt): A preliminary, scenario-based cross-sectional study. *Turkish Journal of Emergency Medicine*, 23(3):156–161.

Jonathon Stewart, Juan Lu, Adrian Goudie, Glenn Arendts, Shiv Akarsh Meka, Sam Freeman, Katie

Walker, Peter Sprivulis, Frank Sanfilippo, Mohammed Bennamoun, et al. 2023. Applications of natural language processing at emergency department triage: A narrative review. *Plos one*, 18(12):e0279953.

Bing Wang, Weizi Li, Anthony Bradlow, Eghosa Bazuaye, and Antoni TY Chan. 2023. Improving triaging from primary care into secondary care using heterogeneous data-driven hybrid machine learning. *Decision Support Systems*, 166:113899.

Bo Yang and Lijun Wu. 2021. How to leverage multimodal ehr data for better medical predictions? *arXiv preprint arXiv:2110.15763*.

A Appendix

Table 3 to Table 12 depict how the generation of the sentences is conducted in the case of *text as feature*. This categorization is in line with medical guidelines and how different indications are defined (e.g., hypertension).

Table 13 to Table 15 provide an overview of the frequency of the three different labels in our dataset.

Table 16 presents the detailed results of Table 1 above. Table 3 to Table 12 present the medical knowledge used to translate non-text features into text features for the extra as-text models.

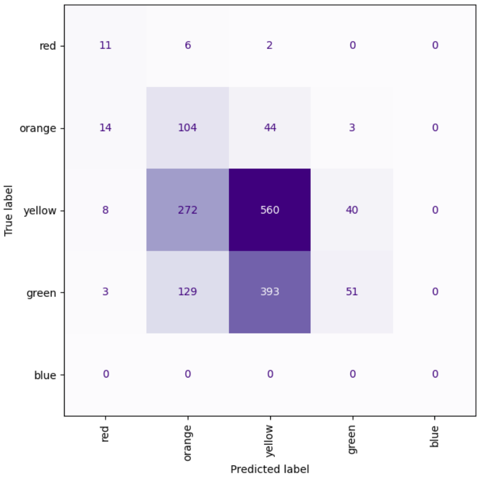


Figure 5: Confusion matrix of top-3 Triage (D) label prediction, where only the most urgent color among the top-3 predictions is counted, from red to blue (very urgent to not urgent).

Table 3: Translation of pain score into a template-based sentence: “Pat. hat ***pain_type*** Schmerzen.” (Pat. has ***pain_type*** pain.)

pain score	0-1	2-3	4-5	6-7	8-9	10
pain_type	“keine” “no”	“leichte” “light”	“mäßige” “moderate”	“starke” “strong”	“sehr starke” “very strong”	“stärkste vorstellbare” “strongest imaginable”

Table 4: Translation of sex value into a template-based sentence: “Pat. ist ***sex_type***.” (Pat. is ***sex_type***.)

sex value	0	1
sex_type	“männlich” “male”	“weiblich” “female”

Table 5: Translation of diastolic value into a template-based sentence: “Pat. hat einen diastolischen Blutdruck von ***X***mmHg.” (Pat. has a diastolic blood pressure of ***X***mmHg.)

diastolic value	X

Table 6: Translation of age into a template-based sentence: “Pat. ist ein ***age_type*** im Alter von ***age***.” (Pat. is a ***age_type*** in the age of ***age***.)

age	≤1	>1 & ≤3	>3 & <18	≥18 & ≤40	>40 & ≤65	>65
age_type	Baby baby	Kleinkind toddler	Kind child	Erwachsener adult	Erwachsener mittleren Alters middle-aged adult	Senior senior

Table 7: Translation of pulse value into a template-based sentence: “Pat. hat einen ***pulse_type*** Puls von ***pulse value***.” (Pat. has a ***pulse_type*** pulse of ***pulse value***.)

pulse value	≤60	>60 & <100	≥100 & ≤120	>120
pulse_type	“zu niedrigen” “too low”	“normalen” “normal”	“erhöhten” “elevated”	“stark erhöhten” “highly elevated”

Table 8: Translation of temperature value into a template-based sentence: “Pat. ist ***temp_type*** mit einer Körpertemperatur von ***temp value*** Grad Celsius.” (Pat. is ***temp_type*** with a body temperature of ***temp value*** degrees Celsius.)

temperature value	≤35	>35 & <37.5	≥37.5 & <38.5	≥38.5 & <41	≥41
temp_type	“unterkühlt” “undercooled”	“normal” “normal”	“überwärmt” “overheated”	“heiß” “hot”	“sehr heiß” “very hot”

Table 9: Translation of spo2 value into a template-based sentence: “Pat. hat eine ***spo2_type*** Sauerstoffsättigung von ***spo2 value***%.” (Pat. has a ***spo2_type*** oxygen saturation of ***spo2 value***%.)

spo2 value	<90	≥90 & <95	≥95
spo2_type	“sehr niedrige” “very low”	“niedrige” “low”	“normal” “normal”

Table 10: Translation of blood sugar value into a template-based sentence: “Pat. hat einen ***bs_type*** Blutzuckerspiegel von ***bs value***mg/dl.” (Pat. has a ***bs_type*** blood sugar level of ***bs value***mg/dl.)

bs value	≤54	>54 & <70	≥70 & ≤100	>100 & <306	≥306
bs_type	“zu niedrigen” “too low”	“niedrigen” “low”	“normalen” “normal”	“erhöhten” “increased”	“zu hohen” “too high”

Table 11: Translation of systolic value into a template-based sentence: “Pat. hat einen ***systolic_type*** systolischen Blutdruck von ***systolic value***mmHg.” (Pat. has a ***systolic_type*** systolic blood pressure of ***systolic value***mmHg.)

systolic value	<90	≥90 & <100	≥100 & ≤120	>120 & ≤140	>140
systolic_type	“zu niedrigen” “too low”	“niedrigen” “low”	“normalen” “normal”	“hohen” “high”	“zu hohen” “too high”

Table 12: Translation of heart frequency value into a template-based sentence: “Pat. hat eine ***hf_type*** Herzfrequenz von ***heart frequency value***.” (Pat. has a ***hf_type*** heart frequency of ***heart frequency value***.)

heart frequency value	<40	≥40 & ≤60	>60 & ≤100	>100 & <140	≥140 & <160	≥160
hf_type	“zu niedrige” “too low”	“niedrige” “low”	“normale” “normal”	“hohe” “high”	“zu hohe” “too high”	“extrem hohe” “extremely high”

Table 13: Frequency of diagram (diagnosis) labels in the dataset

diagram label		# in dataset
Unwohlsein bei Erwachsenen	Discomfort in adults	2480
Stürze	Falls	2104
Extremitätenprobleme	Limb problems	1735
Atemproblem bei Erwachsenen	Respiratory problem in adults	1369
Abdominelle Schmerzen bei Erwachsenen	Abdominal pain in adults	1123
Thoraxschmerz	Thoracic pain	1120
Kopfverletzung	Head injury	808
Urologisches Problem	Urological problem	765
Wunden	Wounds	586
Herzklopfen	Palpitations	554
Kollaps	Collapse	537
Rückenschmerz	Back pain	402
Betrunkener Eindruck	Drunken impression	392
Durchfälle und Erbrechen	Diarrhea and vomiting	244
Generelle Indikatoren	General indicators	234
Gastrointestinale Blutung	Gastrointestinal bleeding	207
Angriff (Zustand nach)	Attack (condition after)	179
Überdosierung und Vergiftung	Overdose and poisoning	143
Körperstammverletzung	Trunk injury	142
Schweres Trauma	Severe trauma	127
Diabetes	Diabetes	118
Nackenschmerz	Neck pain	116
Allergie	Allergy	106
Auffälliges Verhalten	Abnormal behavior	98
Kopfschmerz	Headache	89
Besorgte Eltern	Concerned parents	68
Atemproblem bei Kindern	Breathing problem in children	67
Selbstverletzung	Self-harm	57
Krampfanfall	Seizure	56
Psychiatrische Erkrankung	Psychiatric illness	48
Abdominelle Schmerzen bei Kindern	Abdominal pain in children	45
Unwohlsein bei Kindern	Malaise in children	41
Abszesse und lokale Infektionen	Abscesses and local infections	38
Bisse und Stiche	Bites and stings	32
Verbrennungen und Verbrühungen	Burns and scalds	30
Fremdkörper	Foreign bodies	24
Gesichtsprobleme	Facial problems	24
Asthma	Asthma	21
Hodenschmerz	Testicular pain	20
Halsschmerz	Sore throat	12
Hautausschläge	Skin rashes	9
Unwohlsein bei Neugeborenen	Discomfort in newborns	7
Chemikalienkontakt	Chemical contact	7
Augenprobleme	Eye problems	6
Vaginale Blutung	Vaginal bleeding	3
Unwohlsein bei Säuglingen	Discomfort in infants	2
Ohrenprobleme	Ear problems	2

Table 14: Frequency of indication (discriminator) labels in the dataset

indication label	# in dataset
Recent problem	3931
Moderate pain	3116
Unstoppable minor bleeding	1322
Recent mild pain	1049
Low O2 saturation	625
Rapid onset	579
Report of unconsciousness	377
Abnormal cardiac history	334
Inappropriate history	321
New abnormal pulse	311
Altered state of consciousness can be fully explained by alcohol consumption	267
Cardiac pain	248
Swelling	242
Hot	214
–None	211
Gross misalignment	203
Persistent palpitations	191
Severe pain	152
Very low O2 saturation	151
Tarry stools or fresh blood accumulation	131
Altered state of consciousness	124
Colicky pain	121
Vomiting	118
Urinary retention	118
Conspicuous injury mechanism	108
Tendency to bleed	91
Pleural pain	84
Macrohematuria	74
Shock	71
Overheated	66
Abnormal psychiatric history	63
Wheezing	61
Report of acute vomiting of blood	58
Conspicuous hematological or metabolic anamnesis	58
Fresh neurological deficit	51
Suspected sepsis	50
Moderate risk of (future) self-harm	48
Cannot speak in complete sentences	48
Hyperglycemia	46
Inadequate breathing	42
Signs of dehydration	41
Compromised airway	39
Fresh or old blood stools	38
State of exhaustion	35
High risk of (future) self-harm	33
Moderate pain or itching	33
Conspicuous respiratory history	33
Direct neck trauma	32
Recent injury	32
Scalp hematoma	32
New state of confusion	29
Noticeable restlessness	29
Productive cough	28
Unstoppable major bleeding	27
Local infection	24
Vomiting of blood	23
Malposition	22
Dysuria	20
Unable to walk	19
Acute neurological deficit	19
Hypoglycemia	19
Smoke exposure	19
Local inflammation	18
Hypothermia	18
Recent mild pain or itching	15
Direct back trauma	13
Altered state of consciousness	12
Persistent vomiting	12
Extensive secretions or vesicle formation	11
Inhalation trauma	10
Impaired (distal) circulation	10
Facial edema	10
Scrotal swelling/redness	10
Low peak flow	8
Known or suspected immunosuppression	8
Acute respiratory distress	8
Moderate lethality	8
Report of overdose or intoxication	8
Inadequate history (of alcohol consumption)	7
Hyperglycemia with ketosis	7
Abnormal history of GI bleeding	6
Life-threatening hemorrhage	6
Report of head injury	6
Persistent seizure	5
Tongue edema	5
Electrical accident	5
No response to own asthma medication	5
Radiation of pain into the shoulder	5
Critical skin condition	5
Open fracture	5
Very low peak flow	5
Very hot	4
Pain radiating to the back	4
Overheated joint	3
Stridor	3
Moderately lethal animal bite	3
...	

Table 15: Frequency of triage labels in the dataset

triage label	# in dataset
red	187
orange	1551
yellow	8785
green	5684
blue	190

Table 16: Performance Metrics Top-3

	Diagnose						Indication						Triage (direct)						Triage (indirect)					
	recall		precision		F1		recall		precision		F1		recall		precision		F1		recall		precision		F1	
	w	m	w	m	w	m	w	m	w	m	w	m	w	m	w	m	w	m	w	m	w	m	w	m
MT expert dia	0.621	0.412	0.605	0.455	0.6	0.415	0.344	0.11	0.279	0.201	0.251	0.126	0.587	0.419	0.56	0.432	0.566	0.418	0.541	0.388	0.551	0.363	0.533	0.365
top2	0.795	0.552	0.792	0.587	0.791	0.561	0.541	0.209	0.437	0.302	0.427	0.219	0.884	0.712	0.887	0.639	0.884	0.664	0.509	0.326	0.468	0.358	0.433	0.298
top3	0.861	0.622	0.862	0.625	0.859	0.612	0.646	0.304	0.54	0.386	0.541	0.307	0.976	0.918	0.977	0.833	0.976	0.868	0.502	0.298	0.394	0.34	0.359	0.241
MT expert tri	0.599	0.377	0.571	0.45	0.565	0.394	0.367	0.106	0.291	0.171	0.264	0.111	0.579	0.45	0.573	0.399	0.574	0.415	0.55	0.399	0.567	0.334	0.545	0.351
top2	0.789	0.563	0.772	0.597	0.774	0.561	0.551	0.175	0.466	0.263	0.456	0.188	0.889	0.744	0.896	0.621	0.89	0.667	0.516	0.321	0.485	0.34	0.446	0.291
top3	0.863	0.67	0.856	0.672	0.857	0.658	0.647	0.294	0.572	0.367	0.569	0.294	0.979	0.95	0.979	0.821	0.978	0.871	0.479	0.298	0.426	0.356	0.377	0.262
MT expert & extra (feat.) dia	0.604	0.379	0.601	0.42	0.599	0.393	0.319	0.105	0.255	0.185	0.23	0.115	0.579	0.404	0.541	0.439	0.55	0.413	0.531	0.341	0.54	0.339	0.527	0.335
top2	0.778	0.554	0.779	0.576	0.776	0.551	0.532	0.205	0.425	0.298	0.421	0.214	0.876	0.709	0.876	0.66	0.876	0.681	0.52	0.33	0.484	0.374	0.454	0.311
top3	0.839	0.648	0.839	0.636	0.836	0.631	0.633	0.268	0.527	0.346	0.533	0.27	0.973	0.915	0.973	0.825	0.972	0.862	0.541	0.325	0.428	0.363	0.398	0.267
MT expert & extra (text) tri	0.627	0.405	0.593	0.474	0.591	0.422	0.357	0.106	0.301	0.186	0.269	0.121	0.586	0.418	0.571	0.408	0.575	0.403	0.542	0.41	0.56	0.338	0.535	0.354
top2	0.798	0.526	0.783	0.591	0.785	0.545	0.549	0.177	0.47	0.27	0.453	0.193	0.893	0.727	0.898	0.642	0.894	0.676	0.522	0.355	0.496	0.362	0.453	0.318
top3	0.868	0.611	0.862	0.646	0.862	0.616	0.646	0.267	0.575	0.348	0.568	0.274	0.98	0.944	0.981	0.804	0.98	0.855	0.531	0.337	0.426	0.367	0.384	0.274
MT expert & extra (feat.) disc	0.602	0.39	0.572	0.46	0.575	0.404	0.335	0.113	0.283	0.169	0.27	0.121	0.569	0.397	0.544	0.419	0.549	0.403	0.548	0.409	0.561	0.359	0.548	0.375
top2	0.77	0.508	0.747	0.57	0.751	0.516	0.528	0.198	0.459	0.263	0.46	0.207	0.887	0.75	0.891	0.675	0.888	0.706	0.544	0.364	0.52	0.385	0.495	0.346
top3	0.833	0.578	0.821	0.62	0.822	0.582	0.628	0.277	0.576	0.343	0.581	0.287	0.967	0.92	0.968	0.846	0.967	0.877	0.506	0.318	0.448	0.374	0.411	0.287
MT expert disc	0.547	0.352	0.51	0.416	0.503	0.363	0.353	0.16	0.335	0.179	0.331	0.152	0.561	0.4	0.546	0.396	0.55	0.393	0.551	0.385	0.565	0.357	0.554	0.367
top2	0.709	0.455	0.671	0.519	0.673	0.468	0.517	0.242	0.505	0.252	0.5	0.225	0.869	0.669	0.877	0.6	0.871	0.626	0.529	0.344	0.517	0.358	0.493	0.33
top3	0.782	0.528	0.755	0.565	0.757	0.526	0.639	0.317	0.636	0.31	0.628	0.293	0.953	0.845	0.955	0.807	0.953	0.824	0.532	0.333	0.47	0.353	0.419	0.283
MT extra (feat.) tri	0.605	0.367	0.575	0.442	0.563	0.384	0.349	0.103	0.291	0.18	0.254	0.113	0.582	0.45	0.552	0.402	0.558	0.392	0.527	0.442	0.544	0.314	0.52	0.333
top2	0.774	0.496	0.754	0.554	0.756	0.507	0.544	0.195	0.454	0.27	0.435	0.198	0.895	0.753	0.894	0.66	0.893	0.681	0.521	0.34	0.484	0.335	0.451	0.293
top3	0.855	0.626	0.846	0.65	0.848	0.625	0.632	0.259	0.547	0.337	0.535	0.26	0.983	0.957	0.983	0.818	0.981	0.858	0.519	0.329	0.423	0.354	0.384	0.269
MT expert & extra (feat.) tri	0.602	0.362	0.567	0.445	0.559	0.381	0.361	0.096	0.293	0.173	0.267	0.108	0.574	0.48	0.553	0.384	0.556	0.389	0.556	0.509	0.566	0.331	0.543	0.352
top2	0.788	0.538	0.769	0.59	0.772	0.545	0.541	0.189	0.452	0.272	0.435	0.195	0.888	0.781	0.891	0.629	0.888	0.674	0.517	0.341	0.487	0.347	0.448	0.301
top3	0.86	0.629	0.848	0.649	0.85	0.62	0.627	0.254	0.546	0.337	0.536	0.257	0.982	0.97	0.982	0.826	0.981	0.877	0.501	0.329	0.429	0.356	0.385	0.275
MT expert & extra (text) disc	0.574	0.376	0.538	0.452	0.539	0.392	0.363	0.15	0.327	0.183	0.328	0.157	0.561	0.393	0.548	0.39	0.55	0.389	0.552	0.363	0.565	0.345	0.552	0.35
top2	0.741	0.482	0.716	0.535	0.721	0.493	0.53	0.237	0.5	0.263	0.501	0.234	0.879	0.688	0.885	0.613	0.88	0.642	0.546	0.334	0.519	0.343	0.493	0.311
top3	0.821	0.602	0.81	0.623	0.812	0.6	0.641	0.292	0.625	0.309	0.624	0.28	0.959	0.838	0.96	0.811	0.959	0.819	0.537	0.324	0.461	0.344	0.41	0.267
MT extra (text) disc	0.541	0.34	0.48	0.377	0.477	0.337	0.344	0.159	0.324	0.143	0.323	0.136	0.563	0.422	0.52	0.378	0.52	0.381	0.546	0.425	0.555	0.344	0.549	0.368
top2	0.707	0.44	0.646	0.469	0.655	0.437	0.531	0.266	0.523	0.24	0.514	0.233	0.853	0.679	0.851	0.605	0.846	0.612	0.545	0.408	0.543	0.394	0.537	0.394
top3	0.784	0.527	0.738	0.551	0.747	0.518	0.655	0.344	0.654	0.29	0.64	0.292	0.937	0.839	0.938	0.757	0.935	0.768	0.529	0.336	0.49	0.362	0.445	0.303
MT extra (text) tri	0.615	0.407	0.579	0.48	0.577	0.421	0.348	0.107	0.295	0.169	0.266	0.112	0.582	0.419	0.576	0.368	0.576	0.379	0.55	0.418	0.57	0.32	0.546	0.342
top2	0.8	0.548	0.782	0.594	0.785	0.553	0.549	0.201	0.465	0.263	0.456	0.201	0.884	0.716	0.893	0.583	0.886	0.627	0.521	0.351	0.496	0.358	0.457	0.318
top3	0.87	0.607	0.863	0.642	0.863	0.61	0.653	0.29	0.577	0.33	0.576	0.276	0.979	0.951	0.98	0.801	0.978	0.848	0.509	0.32	0.43	0.35	0.381	0.264
MT expert & extra (text) dia	0.621	0.42	0.615	0.462	0.612	0.428	0.333	0.107	0.257	0.19	0.237	0.119	0.569	0.386	0.546	0.384	0.552	0.378	0.535	0.345	0.552	0.318	0.534	0.325
top2	0.799	0.56	0.802	0.553	0.797	0.541	0.555	0.201	0.438	0.277	0.444	0.208	0.877	0.709	0.884	0.597	0.879	0.635	0.524	0.343	0.493	0.353	0.468	0.316
top3	0.845	0.61	0.848	0.589	0.843	0.589	0.651	0.298	0.557	0.365	0.565	0.299	0.976	0.955	0.976	0.806	0.975	0.859	0.507	0.323	0.427	0.356	0.388	0.274
MT extra (text) dia	0.606	0.403	0.601	0.444	0.6	0.411	0.327	0.095	0.256	0.184	0.232	0.108	0.563	0.404	0.54	0.372	0.545	0.371	0.529	0.335	0.546	0.312	0.529	0.318
top2	0.789	0.541	0.794	0.539	0.788	0.529	0.557	0.185	0.445	0.285	0.446	0.198	0.873	0.707	0.877	0.618	0.874	0.651	0.537	0.365	0.49	0.359	0.462	0.323
top3	0.847	0.641	0.851	0.605	0.845	0.605	0.646	0.262	0.544	0.349	0.551	0.269	0.973	0.942	0.974	0.815	0.973	0.864	0.534	0.333	0.415	0.352	0.384	0.267
MT extra (feat.) dia	0.621	0.405	0.616	0.447	0.613	0.41	0.362	0.107	0.249	0.183	0.229	0.11	0.578	0.421	0.526	0.376	0.529	0.37	0.518	0.347	0.537	0.296	0.519	0.306
top2	0.805	0.58	0.796	0.579	0.793	0.548	0.564	0.196	0.413	0.283	0.421	0.198	0.874	0.725	0.876	0.655	0.874	0.682	0.524	0.368	0.496	0.353	0.467	0.325
top3	0.851	0.661	0.847	0.629	0.843	0.623	0.644	0.246	0.51	0.334	0.524	0.249	0.973	0.922	0.974	0.834	0.973	0.871	0.508	0.339	0.442	0.357	0.409	0.292
MT weighted dia	0.587	0.368	0.598	0.399	0.588	0.377	0.336	0.112	0.249	0.177	0.233	0.113	0.562	0.517	0.549	0.348	0.548	0.369	0.528	0.369	0.549	0.296	0.527	0.31
top2	0.716	0.495	0.727	0.502	0.717	0.485	0.523	0.193	0.396	0.274	0.4	0.194	0.886	0.767	0.895	0.548	0.885	0.595	0.515	0.363	0.501	0.339	0.463	0.314
top3	0.737	0.535	0.747	0.526	0.737	0.517	0.629	0.26	0.519	0.342	0.53	0.26	0.975	0.985	0.974	0.781	0.971	0.836	0.53	0.345	0.426	0.336	0.38	0.264
MT weighted tri	0.62	0.401	0.583	0.464	0.581	0.408	0.337	0.106	0.282	0.185	0.247	0.118	0.561	0.426	0.547	0.358	0.55	0.363	0.54	0.44	0.555	0.314	0.532	0.33
top2	0.79	0.534	0.77	0.579	0.772	0.534	0.552	0.196	0.458	0.278	0.445	0.204	0.885	0.778	0.889	0.597	0.884	0.643	0.526	0.387	0.495	0.341	0.453	0.308
top3	0.856	0.647	0.847	0.657	0.848	0.633	0.648	0.272	0.562	0.35	0.556	0.279	0.98	0.956	0.981	0.794	0.98	0.848	0.519	0.35	0.432	0.35	0.386	0.274
expert	0.617	0.409	0.612	0.453	0.608	0.416	0.358	0.142	0.302	0.178	0.303	0.147	0.571	0.601	0.551	0.348	0.55	0.362	0.541	0.383	0.549	0.375	0.545	0.379
top2	0.807	0.57	0.807	0.548	0.804	0.546	0.53	0.216	0.473	0.253	0.481	0.219	0.904	0.861	0.909	0.542	0.898	0.577	0.52	0.323	0.488	0.369	0.476	0.318
top3	0.863	0.622	0.868	0.583	0.863	0.59	0.648	0.272	0.608	0.297	0.615	0.267	0.983	0.964	0.983	0.77	0.981	0.829	0.499	0.316	0.441	0.386	0.406	0.29
expert & extra (feat.)	0.604	0.388	0.601	0.428	0.595	0.393	0.338	0.121	0.315	0.148	0.316	0.122	0.525	0.417	0.54	0.333	0.529	0.36	0.531	0.356	0.545	0.34	0.537	0.347
top2	0.783	0.522	0.788	0.511	0.782	0.504	0.529	0.213	0.512	0.238	0.51	0.208	0.684	0.699	0.692	0.584	0.682	0.624	0.532	0.341	0.517	0.369	0.497	0.334
top3	0.824	0.572	0.829	0.549	0.822	0.547	0.642	0.285	0.634	0.283	0.628	0.261	0.714	0.776	0.72	0.706	0.71	0.729	0.537	0.322	0.48	0.36	0.425	0.275
weighted	0.611	0.41	0.61	0.426	0.607	0.401	0.344	0.114	0.282	0.16	0.272	0.12	0.541	0.507	0.552	0.325	0.539	0.356	0.522	0.35	0.54	0.318	0.528	0.33
top2	0.743	0.538	0.74	0.52	0.736	0.506	0.546	0.205	0.472	0.244	0.478	0.202	0.688	0.741	0.694	0.572	0.683	0.63	0.528	0.34	0.513	0.354	0.488	0.324
top3	0.774	0.596	0.772	0.572	0.768	0.56	0.662	0.282	0.613	0.303	0.62	0.27	0.709	0.783	0.716	0.731	0.705	0.743	0.518	0.317	0.457	0.357	0.412	0.276
MT weighted disc	0.606	0.4	0.567	0.433	0.576	0.402	0.366	0.138	0.33	0.186	0.325	0.145	0.57	0.4	0.538	0.353	0.542	0.355	0.561	0.429	0.574	0.334	0.564	0.354
top2	0.78	0.512	0.757	0.531	0.764	0.511	0.537	0.217	0.505	0.261	0.505	0.221	0.874	0.711	0.874	0.604	0.869	0.632	0.515	0.348	0.506	0.338	0.491	0.325
top3	0.84	0.578	0.827	0.583	0.831	0.568	0.652	0.285	0.633	0.32	0.633	0.285	0.959	0.852	0.957	0.79	0.956	0.798	0.511	0.341	0.477	0.356	0.436	0.305
extra (text)	0.618	0.403	0.613	0.456	0.607	0.414	0.344	0.124	0.284	0.18	0.279	0.132	0.56	0.393	0.544	0.346	0.542	0.349	0.538	0.344	0.546	0.348	0.533	0.338
top2	0.808	0.576	0.802	0.574	0.798	0.554	0.542	0.196	0.471	0.265	0.472	0.21	0.904	0.863	0.91	0.567	0.899	0.609	0.534	0.325	0.486	0.358	0.454	0.294
top3	0.864	0.663	0.863	0.646	0.858	0.638	0.658	0.285	0.599	0.33	0.604	0.282	0.982	0.957	0.983	0.742	0.979	0.794	0.532	0.321	0.416	0.363	0.377	0.257
expert & extra (text)	0.594	0.388	0.595	0.435	0.589	0.397	0.374	0.12	0.288	0.196	0.267	0.128	0.551	0.385	0.546	0.333	0.545	0.345	0.549	0.332	0.548	0.349	0.542	0.335
top2	0.769	0.546	0.771	0.554	0.764	0.535	0.564	0.192	0.455	0.284	0.456	0.204	0.881	0.708	0.893	0.544	0.882	0.587	0.539	0.322	0.464	0.37	0.446	0.293
top3	0.803	0.601	0.805	0.581	0.797	0.574	0.666	0.258	0.571	0.338	0.578	0.265	0.973	0.964	0.973	0.817	0.972	0.875	0.516	0.297	0.365	0.346	0.341	0.226
normal	0.596	0.404	0.603	0.401	0.592	0.384	0.325	0.113	0.348	0.113	0.33	0.102	0.54	0.411	0.546	0.322	0.54	0.34	0.537	0.386	0.552	0.315	0.542	0.334
top2	0.798	0.58	0.802	0.538	0.794	0.541	0.474	0.222	0.488	0.198	0.472	0.192	0.806	0.684	0.81	0.521	0.802	0.559	0.524	0.352	0.505	0.335	0.492	0.321
top3	0.863	0.648	0.866	0.598	0.86	0.607	0.577	0.284	0.587	0.25	0.572	0.247	0.905	0.881	0.902	0.713	0.9	0.764	0.502	0.306	0.42	0.312	0.392	0.252
MT extra (feat.) disc	0.556	0.352	0.504	0.38	0.511	0.345	0.345	0.112	0.315	0.132	0.316	0.113	0.552	0.386	0.534	0.331	0.533	0.335	0.538	0.341	0.557	0.303	0.544	0.314
top2	0.733	0.462	0.691	0.487	0.703	0.455	0.523	0.191	0.5	0.229	0.501	0.193	0.875	0.72	0.882	0.585	0.872	0.623	0.527	0.345	0.524	0.336	0.506	0.327
top3	0.804	0.552	0.779	0.552	0.785	0.531	0.642	0.303	0.634	0.319	0.631	0.293	0.968	0.953	0.968	0.798	0.966	0.845	0.525	0.333	0.49	0.329	0.44	0.285
extra (feat.)	0.631	0.412	0.593	0.462	0.587	0.415	0.335	0.124	0.252	0.202	0.232	0.133	0.534	0.348	0.538	0.315	0.536	0.325	0.518	0.309	0.521	0.311	0.513	0.305
top2	0.806	0.539	0.793	0.576	0.795	0.546	0.545	0.189	0.418	0.272	0.417	0.201	0.754	0.68	0.76	0.577	0.756	0.617	0.516	0.304	0.458	0.335	0.433	0.273
top3	0.872	0.594	0.87	0.614	0.869	0.594	0.662	0.269	0.548	0.344	0.559	0.278	0.796	0.798	0.8	0.718	0.796	0.748	0.536	0.306	0.38	0.336	0.355	0.228

Creating Ontology-annotated Corpora from Wikipedia for Medical Named-entity Recognition

Johann Frei and **Frank Kramer**
IT-Infrastructure for Translational Medical Research
University of Augsburg
<firstname>.<lastname>@informatik.uni-augsburg.de

Abstract

Acquiring annotated corpora for medical NLP is challenging due to legal and privacy constraints and costly annotation efforts, and using annotated public datasets may do not align well to the desired target application in terms of annotation style or language. We investigate the approach of utilizing Wikipedia and WikiData jointly to acquire an unsupervised annotated corpus for named-entity recognition (NER). By controlling the annotation ruleset through WikiData's ontology, we extract custom-defined annotations and dynamically impute weak annotations by an adaptive loss scaling. Our validation on German medication detection datasets yields competitive results. The entire pipeline only relies on open models and data resources, enabling reproducibility and open sharing of models and corpora. All relevant assets are shared on GitHub[1].

1 Introduction

A major reoccurring pain point in natural language processing (NLP) remains the issue of lacking resources regarding text corpora, including adequate annotation information in particular. Especially in medical and clinical NLP environments, the situation is notoriously difficult due to privacy and legal restrictions on data use and sharing; rendering efforts to share open datasets from the clinical domain complex and cumbersome. Yet there are notable and important attempts to provide text resources close to the clinical domain to the public research domain, e.g. MIMIC-III/-IV (Pollard and Johnson, 2016; Johnson et al., 2023). While these resources are extremely valuable, a single annotated dataset is governed by pre-defined parameters: First, the actual language, which may not align with the desired target language, and may pose another challenge regarding cross-lingual transfer. Second, the domain-dependent linguistic text properties and styles may vary from corpus to corpus. Third, the provided annotation data are usually tied to a certain ontology in entity linking, or label classes in named-entity recognition (NER). Forth, even if the label classes appear identical across multiple corpora, the underlying annotation guidelines that were employed as rulesets for annotation decision-making are usually not identical or consistent with annotation guidelines from other datasets. While the language and the text style are inherently fixed, new annotation data for a given corpus could be manually created if a custom annotation guideline is needed. However, a manual annotation is costly, resource-intensive and may remain non-reproducible to a certain degree due to the human-provided input. Hence, creating such an alternative annotation layer is not feasible in practice. In this work, we investigate the practicality of applying a fully unsupervised approach for annotated data acquisition in the medical context, yielding a corpus that is subsequently used as training material for an NER model. To achieve this, we compose several steps to obtain our final results. Our approach combines two public knowledge sources, Wikipedia[2] and WikiData (Vrandečić and Krötzsch, 2014)[3], to extract text data and annotation information, whereas the core annotation ruleset can be defined by leveraging the graph-like ontology structure of WikiData. The crafted dataset is used to train a conventional NER model. To demonstrate the feasibility of our approach, we evaluate our trained NER models on several external public datasets. Since our approach is in particular of interest for medium- to low-resource languages, we choose German as our non-English target language, but it is also motivated by the fact that external annotated datasets are available in that

[1] https://github.com/frankkramer-lab/WikiOntoNERCorpus

[2] https://www.wikipedia.org/ (accessed July 5th, 2024)

[3] https://www.wikidata.org/ (accessed July 5th, 2024)

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 570–579
August 16, 2024. ©2024 Association for Computational Linguistics

language for a final evaluation.

Due to its simple availability, quality and text size, Wikipedia has been subject to NLP research in the past decade, in particular exploited further to obtain NER corpora. Therefore, it has been applied in numerous works for generic NER (Ghaddar and Langlais, 2017; Ryu et al., 2017; Nothman et al., 2008; Nothman et al., 2013; Hahm et al., 2014; Kim et al., 2012; Richman and Schone, 2008; Ni and Florian, 2016; Krishnan et al., 2021; Tsai et al., 2016; Alves et al., 2021), mostly using Wikipedia text or features, while others also include structured knowledge bases like DBpedia (Mendes et al., 2012) or FreeBase (Bollacker et al., 2008). Similar to our work, Jiang et al. 2021 also cover English biomedical NER for weak annotation data by modifying the loss function to be "noise-aware" and applying annotation imputation. Yet they also include a set of small, manually fully-annotated labels, and use PubMed texts with automatic dictionary-based label synthesis instead of Wikipedia resources. Regarding open domain NER, Liang et al. 2020 tackle the challenge by a similar two-stage self-training approach using WikiData, but do not further cover any custom Wikipedia parsing. To the best of our knowledge, no work using Wikipedia and WikiData has been reported so far in the German, medical domain.

2 Methods

2.1 Mapping Wikipedia and WikiData

Throughout this work, we consider Wikipedia as a language-dependent set of text documents identified by unique titles. The text documents are encoded in WikiText, a domain-specific language in markup style, which we mainly treat as plain text sequences along with span-oriented text references to other Wikipedia documents. In contrast, WikiData is a language-agnostic knowledge base which encodes its knowledge in a graph structure with typed, directed edges ("statements") between individual nodes. Each node is a WikiData entity, uniquely identified by its QID number, and either represents an actual concept (e.g. *cancer* (Q12078)) and therefore may be referenced to its corresponding language-specific Wikipedia page, or is part of a virtual concept that encodes certain ontology-inspired hierarchy structures (e.g. *class of disease* (Q112193867)). Note that the correspondences between the WikiData entities to their language-specific Wikipedia pages are bijective in most cases. We utilize these references to establish a mapping between WikiData and Wikipedia entries.

2.2 Extracting Annotations from Wikipedia

The WikiText markup language, which is used to encode the Wikipedia pages, facilitates the use of references to other Wikipedia pages from the same language. These kinds of references are eventually rendered as hyperlinks in web browsers, and are used to link certain terms within a Wikipedia page text to pages that address the mentioned concepts as their main topic. Given a set of concepts we are interested in, defined as a set of Wikipedia pages in practice, we parse the language-specific Wikipedia dump to extract all sentences that contain references to our set of concepts of interest while we retain the reference information of each of the extracted sentences with regard to the text span of the link and its target page. Note that for each extracted sentence, we also keep the information on references that were *not* part of our set of concepts as *negative* mentions. Finally, we obtain an annotated corpus in a certain language that contains annotation information for mentions of concepts of our interest. However, since not every mentioned concept is usually referenced in the WikiText and concepts worth referencing are usually only reference at the first occurrence on a page, the obtained corpus only contains weakly-annotated labels. Using the corpus as training resource to directly train an NER model therefore is expected to yield a model with high precision, yet very low recall scores due to the weak annotation.

2.3 Graph-defined Entity Selection using SPARQL

To enable the use of the WikiData ontology to define a set of concepts of interest, we leverage the SPARQL interface[4], an RDF query language, to determine all WikiData entities of interest. By these means, we can make use of more complex queries that take full advantage of the WikiData ontology structures, and thus it yields an explainable and well-defined output. We further resolve the WikiData entities into their language-specific Wikipedia pages by the mapping we established before, and extract all relevant sentences with annotations, as described earlier. The entire process is illustrated in Figure 1.

[4] https://query.wikidata.org/ accessed May 3rd, 2024

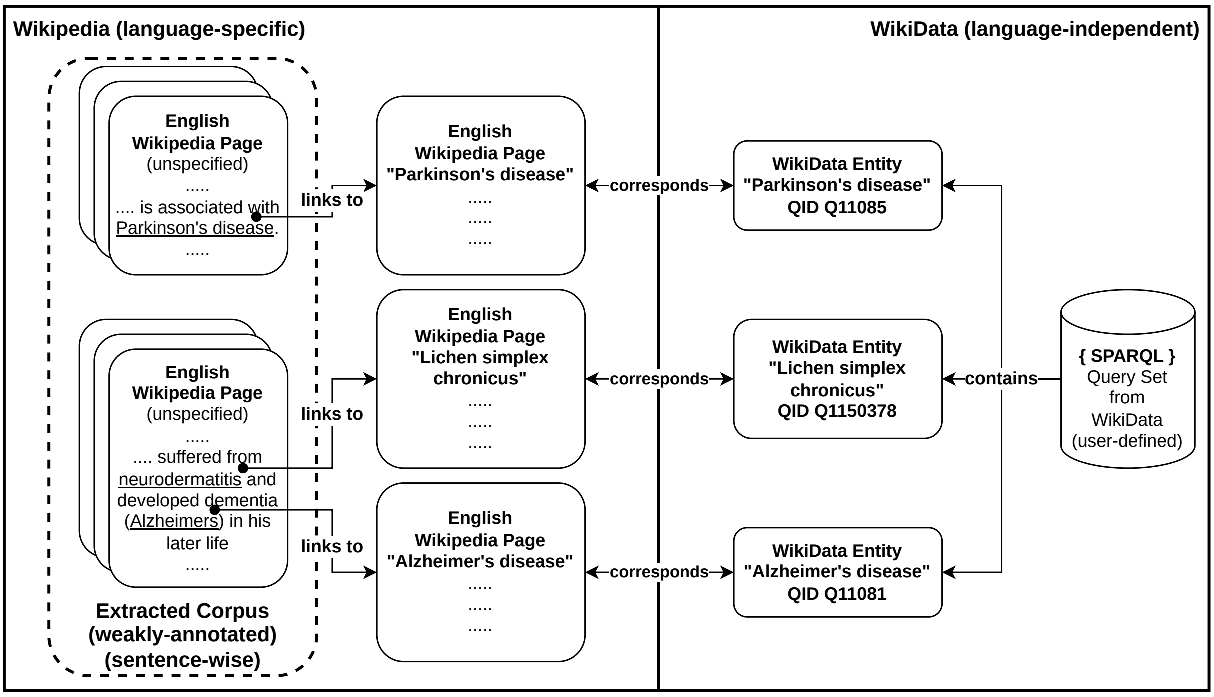

Wikipedia (language-specific)
WikiData (language-independent)
English Wikipedia Page (unspecified) .... is associated with Parkinson's disease. .....
links to
English Wikipedia Page "Parkinson's disease" ....
corresponds
WikiData Entity "Parkinson's disease" QID Q11085
English Wikipedia Page "Lichen simplex chronicus" ....
WikiData Entity "Lichen simplex chronicus" QID Q1150378
contains
{ SPARQL } Query Set from WikiData (user-defined)
English Wikipedia Page (unspecified) .... suffered from neurodermatitis and developed dementia (Alzheimers) in his later life .....
links to
links to
English Wikipedia Page "Alzheimer's disease" ....
corresponds
corresponds
WikiData Entity "Alzheimer's disease" QID Q11081
Extracted Corpus (weakly-annotated) (sentence-wise)

Figure 1: Conceptual design of the weakly-annotated dataset creation for a certain SPARQL query on WikiData.

2.4 Dataset Imputation of Weak Annotations

Since NER can be framed as a token classification task, we can stratify our evidence about a token class into three cases: First, if the token is an actual part of a reference to a page from our set of concepts, it is considered a *positive* token. Second, if the token is also part of a reference, but does not link to our desired concept set, we consider it a *negative* token. Any other token without any reference is declared as *unknown*, for which we assume the token to not belong to any named entity class similar to *negative* tokens, yet due to the weak annotation its actual class remains unclear. To mitigate the weak label signal during training, we dynamically scale the gradient loss for each token by the factor ω according to the following schema:

$$L_{scaled} = \begin{cases} \omega_{pos} L & \text{if token} \in \text{positive} \\ \omega_{neg} L & \text{if token} \in \text{negative} \\ \omega_{unk} L & \text{if token} \in \text{unknown} \end{cases} \quad (1)$$

We balance the loss scaling weights ω accordingly.

$$\omega_{pos} = \frac{\#tokens_{neg}}{\#tokens_{pos}}, \omega_{neg} = \frac{\#tokens_{pos}}{\#tokens_{neg}} \quad (2)$$

ω_{unk} remains choosable as a hyperparameter. Given the actual positive tokens, the negative tokens serving as contrastive samples, as well as the unknown token samples, we can train an NER model while maintaining the dynamic loss scaling at each token position. Given the trained NER model, it can be re-applied to the weakly-annotated corpus in order to impute missing annotation spans to subsequently obtain a silver standard, fully-annotated corpus as shown in Figure 2.

3 Results

Regarding our implementation, significant portions are re-purposed from existing work (Frei et al., 2022). To assess our proposed approach in medical, non-English NER, we mimic a medication detection task in German texts due to the availability of public datasets with annotations including label classes semantically related to drug or medication, namely BRONCO150 (Kittner et al., 2021), CARDIO:DE (Richter-Pechanski et al., 2023), GPTNERMED (Frei and Kramer, 2023), GERNERMED++ (Frei et al., 2023), and GGPOnc 2 (Borchert et al., 2022) (with short, fine annotation layer). To address the medication detection task, our simple entity selection strategy hereby filters all WikiData entries that have an ATC code assigned through the WikiData property *P267* to eventually obtain a weakly-annotated corpus. Based on this corpus, we fine-tune an NER model with the Huggingface Transformers (Wolf et al., 2020) library for different ω_{unk} scalars while $\omega_{pos}/\omega_{neg}$

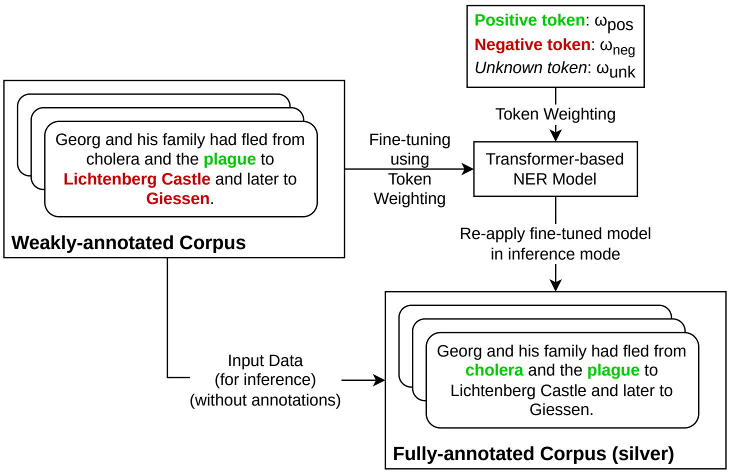

Positive token: ωpos
Negative token: ωneg
Unknown token: ωunk
Token Weighting
Georg and his family had fled from cholera and the plague to Lichtenberg Castle and later to Giessen.
Weakly-annotated Corpus
Fine-tuning using Token Weighting
Transformer-based NER Model
Re-apply fine-tuned model in inference mode
Input Data (for inference) (without annotations)
Georg and his family had fled from cholera and the plague to Lichtenberg Castle and later to Giessen.
Fully-annotated Corpus (silver)

Figure 2: Illustration of the dataset imputation process for the annotation data using dynamic token loss scaling.

remains balanced (1.337/0.748). In order to retain a generic, non-medical setup for the dataset imputation, we use GottBERT (Scheible et al., 2020), a non-domain-specific German RoBERTa model as encoder model for our NER token classifier. To obtain a fully-annotated corpus, we apply the dataset imputation on the weakly-annotated corpus using the fine-tuned NER model. The statistics of the corpus in various configurations are provided in Table 1. Finally, we simulate a traditional but domain-aware NER fine-tuning setup that assumes an ordinary, fully-annotated corpus by training on the imputed corpus and use the default NER training setup of SpaCy (Montani et al., 2023) with GerMedBERT's *medBERT.de* (Bressem et al., 2024), a German medical language model, as BERT encoder without applying any token loss scaling. The scores on the external datasets are evaluated using *BratEval*[5] in overlap mode, and shown in Table 2. The results for all ω_{unk} values are provided in Table 3 in the appendix.

The results on the BRONCO150, GERNERMED++, and GPTNERMED datasets indicate best and robust F1 scores in cases of $\omega_{unk} < 0.8$, whereas for CARDIO:DE the scores are evidently rather modest, to rather poor in case of GGPOnc2. The impact of the token loss scaling ω_{unk} is, as expected, clearly noticeable as it consistently increases the recall rates at lower ω_{unk} values at the expense of minor precision losses. A preliminary lookup into the annotation disagreements in CARDIO:DE and GGPOnc reveals that the models tend to perform poorly on corpora with sparse, less frequent annotations, especially on samples without any annotation. However, another apparent factor remains the conceptual disagreements, for instance "Thrombozyten" was detected due to its assigned ATC code in its WikiData entity *Q101026*, however it was not annotated by the Gold standard annotation.

Property	Value
#Samples	84478
#Sents	90918
#Tokens	2023187
#Labels (pos), raw	105207
#Labels (neg), raw	145492
#Labels (pos), ω_{unk} = 0.01	135795
#Labels (pos), ω_{unk} = 0.05	127950
#Labels (pos), ω_{unk} = 0.1	128157
#Labels (pos), ω_{unk} = 0.2	120973
#Labels (pos), ω_{unk} = 0.5	114339
#Labels (pos), ω_{unk} = 0.8	110237
#Labels (pos), ω_{unk} = 1.0	110024

Table 1: Statistics of the obtained datasets for different ω_{unk} settings. The corpora are based on the SPARQL query that identifies all WikiData entities with assigned ATC codes. The query is given in the appendix.

4 Discussion and Limitations

While a conclusive verdict is not feasible based on the pure evaluation scores due to the diverse and incoherence issues of the external datasets, the fact that the entire data process operates solely on open data yet can perform surprisingly well, even on external corpora, encourages further efforts to foster open NLP resources and models, especially for more sparse, non-English domains. Comparing our results with related work, the GGPOnc 2 baseline

[5] https://github.com/READ-BioMed/brateval/tree/v0.3.2 (Commit c4f5fff) accessed May 3rd, 2024

ω_{unk}	Dataset	Pr	Re	F1
0.01	BRONCO150	0.8103	0.7505	0.7792
0.2	(Kittner et al., 2021)	0.8014	0.7538	0.7768
1.0	`[MEDICATION]`	0.8537	0.5983	0.7035
0.01	GERNERMED++	0.8104	0.7897	0.7999
0.2	(Frei et al., 2023)	0.8453	0.7526	0.7963
1.0	`[Drug]`	0.8831	0.6841	0.771
0.01	GPTNERMED	0.8002	0.8802	0.8383
0.2	(Frei and Kramer, 2023)	0.8553	0.8537	0.8545
1.0	`[Medikation]`	0.8336	0.8172	0.8253
0.01	CARDIO:DE	0.5402	0.7266	0.6197
0.2	(Richter-Pechanski et al., 2023)	0.5352	0.7107	0.6106
1.0	`[DRUG, ACTIVEING]`	0.5634	0.5924	0.5775
0.01	GGPOnc 2	0.1908	0.7257	0.3021
0.2	(Borchert et al., 2022)	0.2324	0.6635	0.3442
1.0	`[Clinical_Drug]` (short, fine)	0.2425	0.5702	0.3402

Table 2: Performance scores on external datasets using BratEval in *overlap* mode for **Pr**ecision, **Re**call and **F1** score for different ω_{unk} values. See Table 3 in the appendix for all ω_{unk} values. The harmonized label classes are given in square brackets.

NER model is reported to achieve .91 F1-score on its test set on the *Clinical_Drug* label class, likewise CARDIO:DE achieves .85/0.81 F1-scores for the *ACTIVEING/DRUG* label classes. However, major disagreements are reported in cross-corpus model transfer. For instance, (Richter-Pechanski et al., 2023) report a .21 F1-score for the DRUG class when applying the GGPOnc 2 NER model to the held-back part of the CARDIO:DE corpus, highlighting certain innate limitations when comparing F1-scores across different datasets and annotation guidelines, as well as the need for the use of multiple datasets during evaluation. As for another, less severe instance, (Frei et al., 2023) and (Frei and Kramer, 2023) report .73/.72. F1-scores respectively on the BRONCO150 corpus for the *MEDICATION* label class, hinting towards more consistent mutual annotation agreements.

Other factors in NER are not further addressed in this work, such as efforts towards support for nested entities or discontinuous annotations or the support for label classes beyond medication detection. The latter aspect may be achieved by the use of an updated SPARQL query definition for certain label classes that align well to the annotation schema from Wikipedia articles but may fail for other label classes like *strength-* or *frequency-*related entities which may not be covered well by Wikipedia or WikiData. In this regard, potential limitations within the WikiData knowledge base are not investigated that may influence the quality of our results in other domains. Other limiting factors such as the quality of the pre-trained language models are not quantified in isolation. However, in general, the effective use of more sophisticated SPARQL queries for entity selection may unlock further potential gains, as well as its application in other languages and domains since our method is not inherently bound to the medical field. Yet, these aspects only serve as motivation for future work.

5 Conclusion

In this work, we demonstrated an unsupervised approach for creating an annotated dataset for medical NER for the German language defined by the WikiData ontology structure using exclusively open data resources. The proof of concept of the proposed method in practical scenarios was shown on a set of external datasets, yielding surprisingly well results. We also discussed further potential but currently underexplored factors such as improved SPARQL queries as future work. Relevant assets are published on GitHub[6], including a web interface that enables external users to create new corpora from custom SPARQL queries for independent experiments.

Acknowledgments

We want to thank Dr. Lisa Raithel for proofreading the manuscript and for providing valuable feedback and support. We also thank all three anonymous reviewers for their constructive and positive feedback.

References

Diego Alves, Gaurish Thakkar, and Marko Tadic. 2021. Building and evaluating universal named-entity recognition english corpus. In *Proceedings of the 2nd International Workshop on Cross-lingual Event-centric Open Analytics co-located with the 30th The Web Conference (WWW 2021), Ljubljana, Slovenia, April 12, 2021 (online event due to COVID-19 outbreak)*, volume 2829 of *CEUR Workshop Proceedings*, pages 2–16. CEUR-WS.org.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, SIGMOD '08, pages 1247–1250. Association for Computing Machinery.

Florian Borchert, Christina Lohr, Luise Modersohn, Jonas Witt, Thomas Langer, Markus Follmann,

[6] `https://github.com/frankkramer-lab/WikiOntoNERCorpus` (accessed July 5th, 2024)

Matthias Gietzelt, Bert Arnrich, Udo Hahn, and Matthieu-P. Schapranow. 2022. GGPONC 2.0 - the german clinical guideline corpus for oncology: Curation workflow, annotation policy, baseline NER taggers. In *Proceedings of the Language Resources and Evaluation Conference*, pages 3650–3660. European Language Resources Association.

Keno K. Bressem, Jens-Michalis Papaioannou, Paul Grundmann, Florian Borchert, Lisa C. Adams, Leonhard Liu, Felix Busch, Lina Xu, Jan P. Loyen, Stefan M. Niehues, Moritz Augustin, Lennart Grosser, Marcus R. Makowski, Hugo JWL Aerts, and Alexander Löser. 2024. MEDBERT.de: A comprehensive german BERT model for the medical domain. *Expert Systems with Applications*, 237:121598.

Johann Frei, Ludwig Frei-Stuber, and Frank Kramer. 2023. GERNERMED++: Semantic annotation in german medical NLP through transfer-learning, translation and word alignment. *Journal of Biomedical Informatics*, 147:104513.

Johann Frei and Frank Kramer. 2023. Annotated dataset creation through large language models for non-english medical NLP. *Journal of Biomedical Informatics*, 145:104478.

Johann Frei, Iñaki Soto-Rey, and Frank Kramer. 2022. Drnote: An open medical annotation service. *PLOS Digital Health*, 1(8):1–18.

Abbas Ghaddar and Phillippe Langlais. 2017. WiNER: A wikipedia annotated corpus for named entity recognition. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 413–422. Asian Federation of Natural Language Processing.

Younggyun Hahm, Jungyeul Park, Kyungtae Lim, Youngsik Kim, Dosam Hwang, and Key-Sun Choi. 2014. Named entity corpus construction using wikipedia and DBpedia ontology. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2565–2569. European Language Resources Association (ELRA).

Haoming Jiang, Danqing Zhang, Tianyu Cao, Bing Yin, and Tuo Zhao. 2021. Named entity recognition with small strongly labeled and large weakly labeled data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1775–1789. Association for Computational Linguistics.

Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, Liwei H. Lehman, Leo A. Celi, and Roger G. Mark. 2023. MIMIC-IV, a freely accessible electronic health record dataset. *Scientific Data*, 10(1):1. Publisher: Nature Publishing Group.

Sungchul Kim, Kristina Toutanova, and Hwanjo Yu. 2012. Multilingual named entity recognition using parallel data and metadata from wikipedia. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 694–702. Association for Computational Linguistics.

Madeleine Kittner, Mario Lamping, Damian T Rieke, Julian Götze, Bariya Bajwa, Ivan Jelas, Gina Rüter, Hanjo Hautow, Mario Sänger, Maryam Habibi, Marit Zettwitz, Till de Bortoli, Leonie Ostermann, Jurica Ševa, Johannes Starlinger, Oliver Kohlbacher, Nisar P Malek, Ulrich Keilholz, and Ulf Leser. 2021. Annotation and initial evaluation of a large annotated german oncological corpus. *JAMIA Open*, 4(2):ooab025.

Aravind Krishnan, Stefan Ziehe, Franziska Pannach, and Caroline Sporleder. 2021. Employing wikipedia as a resource for named entity recognition in morphologically complex under-resourced languages. In *Proceedings of the 14th Workshop on Building and Using Comparable Corpora (BUCC 2021)*, pages 28–39. INCOMA Ltd.

Chen Liang, Yue Yu, Haoming Jiang, Siawpeng Er, Ruijia Wang, Tuo Zhao, and Chao Zhang. 2020. BOND: BERT-assisted open-domain named entity recognition with distant supervision. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, pages 1054–1064. Association for Computing Machinery.

Pablo Mendes, Max Jakob, and Christian Bizer. 2012. DBpedia: A multilingual cross-domain knowledge base. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 1813–1817. European Language Resources Association (ELRA).

Ines Montani, Matthew Honnibal, Matthew Honnibal, Adriane Boyd, Sofie Van Landeghem, and Henning Peters. 2023. explosion/spaCy: v3.7.2: Fixes for APIs and requirements.

Jian Ni and Radu Florian. 2016. Improving multilingual named entity recognition with wikipedia entity type mapping. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1275–1284. Association for Computational Linguistics.

Joel Nothman, James R. Curran, and Tara Murphy. 2008. Transforming wikipedia into named entity training data. In *Proceedings of the Australasian Language Technology Association Workshop 2008*, pages 124–132.

Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R. Curran. 2013. Learning multilingual named entity recognition from wikipedia. *Artificial Intelligence*, 194:151–175.

Tom J Pollard and Alistair EW Johnson. 2016. The MIMIC-III clinical database.

Alexander E. Richman and Patrick Schone. 2008. Mining wiki resources for multilingual named entity recognition. In *Proceedings of ACL-08: HLT*, pages 1–9. Association for Computational Linguistics.

Phillip Richter-Pechanski, Philipp Wiesenbach, Dominic M. Schwab, Christina Kiriakou, Mingyang He, Michael M. Allers, Anna S. Tiefenbacher, Nicola Kunz, Anna Martynova, Noemie Spiller, Julian Mierisch, Florian Borchert, Charlotte Schwind, Norbert Frey, Christoph Dieterich, and Nicolas A. Geis. 2023. A distributable german clinical corpus containing cardiovascular clinical routine doctor's letters. *Scientific Data*, 10(1):207. Publisher: Nature Publishing Group.

Seonghan Ryu, Hwanjo Yu, and Gary Geunbae Lee. 2017. Two-stage approach to named entity recognition using wikipedia and DBpedia. In *Proceedings of the 11th International Conference on Ubiquitous Information Management and Communication*, IMCOM '17, pages 1–4. Association for Computing Machinery.

Raphael Scheible, Fabian Thomczyk, P. Tippmann, V. Jaravine, and M. Boeker. 2020. GottBERT: a pure german language model. *ArXiv*.

Chen-Tse Tsai, Stephen Mayhew, and Dan Roth. 2016. Cross-lingual named entity recognition via wikification. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 219–228. Association for Computational Linguistics.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45. Association for Computational Linguistics.

A Complete NER Results

The results for all ω_{unk} values are provided in Table 3.

ω_{unk}	Dataset	Pr	Re	F1
0.01	BRONCO150 (Kittner et al., 2021) [MEDICATION]	0.8103	0.7505	0.7792
0.05		0.8209	0.7302	0.7729
0.1		0.7762	0.7727	0.7745
0.2		0.8014	0.7538	0.7768
0.5		0.8381	0.6728	0.7464
0.8		0.8618	0.5865	0.698
1.0		0.8537	0.5983	0.7035
0.01	GERNERMED++ (Frei et al., 2023) [Drug]	0.8104	0.7897	0.7999
0.05		0.8363	0.7383	0.7843
0.1		0.7627	0.7654	0.764
0.2		0.8453	0.7526	0.7963
0.5		0.8518	0.7152	0.7776
0.8		0.8773	0.6876	0.771
1.0		0.8831	0.6841	0.771
0.01	GPTNERMED (Frei and Kramer, 2023) [Medikation]	0.8002	0.8802	0.8383
0.05		0.8286	0.8638	0.8458
0.1		0.741	0.8787	0.804
0.2		0.8553	0.8537	0.8545
0.5		0.83	0.8413	0.8356
0.8		0.8145	0.8151	0.8148
1.0		0.8336	0.8172	0.8253
0.01	CARDIO:DE (Richter-Pechanski et al., 2023) [DRUG, ACTIVEING]	0.5402	0.7266	0.6197
0.05		0.5219	0.6774	0.5895
0.1		0.5786	0.7278	0.6447
0.2		0.5352	0.7107	0.6106
0.5		0.5856	0.6506	0.6163
0.8		0.6262	0.5731	0.5985
1.0		0.5634	0.5924	0.5775
0.01	GGPOnc 2 (Borchert et al., 2022) [Clinical_Drug] (short, fine)	0.1908	0.7257	0.3021
0.05		0.2386	0.6944	0.3552
0.1		0.1855	0.7026	0.2935
0.2		0.2324	0.6635	0.3442
0.5		0.2085	0.6265	0.3128
0.8		0.2143	0.5805	0.3131
1.0		0.2425	0.5702	0.3402

Table 3: Performance scores on external datasets using BratEval in *overlap* mode for **Pr**ecision, **Re**call and **F1** score for all different ω_{unk} values. The harmonized label classes are given in square brackets.

B Setup Parameters

B.1 SPARQL Query for Entity Selection

The SPARQL query that has been used for the entity selection. The query selects all WikiData entities with an assigned ATC code.

```
# Anything that has an assigned ATC code
SELECT ?item
WHERE
{
    ?item wdt:P267 ?atccode .
}
```

The query was performed on the WikiData SPARQL query service[7] on April 17th, 2024.

B.2 Training Configuration

B.2.1 First-stage NER with Dynamic Loss Scaling

We use the Transformers (Wolf et al., 2020) library to train the first NER model used for dataset imputation during successive steps. The entire dataset (for ATC) was split into train+dev set (90%) and test set (10%), and the train+dev set was split into train set (80%) and dev set (20%). The following Huggingface Transformers parameters were used for training.

```
"evaluation_strategy": "epoch",
"per_device_train_batch_size": 32,
"per_device_eval_batch_size": 32,
"gradient_accumulation_steps": 1,
"learning_rate": 5e-05,
"weight_decay": 0.0,
"adam_beta1": 0.9,
"adam_beta2": 0.999,
"adam_epsilon": 1e-08,
"max_grad_norm": 1.0,
"num_train_epochs": 3,
"lr_scheduler_type": "linear",
"warmup_ratio": 0.0,
"warmup_steps": 0,
"save_strategy": "epoch",
"seed": 42,
"load_best_model_at_end": true,
"metric_for_best_model": "loss",
"optim": "adamw_torch",
```

As a pre-trained encoder, we used `uklfr/gottbert-base` (Scheible et al., 2020) from the Huggingface Hub. The final model was picked according to the lowest loss on the dev set.

B.2.2 Output Token Decoding for Dataset Imputation

For the decoding of the output probabilities from the first-stage NER model for the dataset imputation, we used a greedy decoding strategy to predict the IOB2 labels (*O*, *B-LABEL*, *I-LABEL*). However, invalid outputs were set to -inf prior to the final token decoding.

B.2.3 Second-stage NER on Imputed Dataset

For the training on the imputed dataset using SpaCy (Montani et al., 2023), the initial default configuration was created with the CLI command:

[7] https://query.wikidata.org/

python3 -m spacy init config base.cfg -l de -p ner -G -o accuracy. The following modifications were made to the base configuration:

- In [components.transformer.model], set name = "GerMedBERT/medbert-512"
- In [training.optimizer.learn_rate], set initial_rate = 5e-5
- In [training], set max_epochs = 10
- In [training], set max_steps = -1
- In [training], set seed = 0

The final configuration was created with the CLI command: python3 -m spacy init fill-config base.cfg final.cfg. Similar to the first-stage NER training, the entire dataset was split into train+dev set (90%) and test set (10%), and the train+dev set was split into train set (80%) and dev set (20%). After training, the best model was picked according to the best (internal) F1 score on the dev set, as this is the default SpaCy approach.

C Data Versioning

C.1 NLP Tools

The Python libraries from pypi.org in the following versions were used for the experiments:

- **Huggingface Transformers**: transformers: 4.36.2
- **SpaCy**: spacy: 3.7.4
- **SpaCy-Transformers**: spacy-transformers: 1.3.4

C.2 Wikipedia and WikiData Dumps

The dumps for WikiData and Wikipedia were accessed by the web references at the following time:

- **Wikipedia / German**: https://dumps.wikimedia.org/dewiki/latest/dewiki-latest-pages-meta-current.xml.bz2 on February 22, 2024.
- **Wikipedia / English**: https://dumps.wikimedia.org/enwiki/latest/enwiki-latest-pages-meta-current.xml.bz2 on February 26, 2024.
- **Wikipedia / French**: https://dumps.wikimedia.org/frwiki/latest/frwiki-latest-pages-meta-current.xml.bz2 on February 26, 2024.
- **Wikipedia / Spanish**: https://dumps.wikimedia.org/eswiki/latest/eswiki-latest-pages-meta-current.xml.bz2 on February 26, 2024.
- **WikiData**: https://dumps.wikimedia.org/wikidatawiki/entities/ on February 23, 2024.

D Annotated Text Samples

To visualize the effect of the annotation imputation stage, several samples from the datasets are shown in Table 4. The datasets are based on the SPARQL query which applies the ATC code assignment filter. While in most instances, the added entities can be considered correct, some ambiguities persist even after manual inspection. For instance, the words "calcium-" and "magnesiumhaltigen" may refer to "calcium carbonate" (Q23767) and "magnesium carbonate" (Q407931) and ATC codes are assigned to their corresponding WikiData entities. However, the correspondent WikiData items for "calcium" (Q706) and "magnesium" (Q660) lack any ATC code.

Setup	Text Sample
raw (neg)	Dabei wird das Kollagen des Fleischbindegewebes durch **Säuren**, Tannine und weitere Bestandteile des Weins angegriffen, gelockert und teilweise gelatiniert, wodurch das Fleisch zarter wird und Geschmack freigesetzt wird.
raw (pos)	Dabei wird das **Kollagen** des Fleischbindegewebes durch Säuren, Tannine und weitere Bestandteile des Weins angegriffen, gelockert und teilweise gelatiniert, wodurch das Fleisch zarter wird und Geschmack freigesetzt wird.
ω_{unk} = 0.01 (imp)	Dabei wird das **Kollagen** des Fleischbindegewebes durch Säuren, Tannine und weitere Bestandteile des Weins angegriffen, gelockert und teilweise **gelatiniert**, wodurch das Fleisch zarter wird und Geschmack freigesetzt wird.
ω_{unk} = 0.2 (imp)	Dabei wird das **Kollagen** des Fleischbindegewebes durch Säuren, Tannine und weitere Bestandteile des Weins angegriffen, gelockert und teilweise **gel**atiniert, wodurch das Fleisch zarter wird und Geschmack freigesetzt wird.
ω_{unk} = 1.0 (imp)	Dabei wird das **Kollagen** des Fleischbindegewebes durch Säuren, Tannine und weitere Bestandteile des Weins angegriffen, gelockert und teilweise gelatiniert, wodurch das Fleisch zarter wird und Geschmack freigesetzt wird.
raw (neg)	Die klinische Entwicklung bei Depression wurde jedoch eingestellt, da Rolipram im Vergleich zu herkömmlichen Antidepressiva keinen Zusatznutzen zeigen konnte.
raw (pos)	Die klinische Entwicklung bei Depression wurde jedoch eingestellt, da Rolipram im Vergleich zu herkömmlichen **Antidepressiva** keinen Zusatznutzen zeigen konnte.
ω_{unk} = 0.01 (imp)	Die klinische Entwicklung bei Depression wurde jedoch eingestellt, da **Rolipram** im Vergleich zu herkömmlichen **Antidepressiva** keinen Zusatznutzen zeigen konnte.
ω_{unk} = 0.2 (imp)	Die klinische Entwicklung bei Depression wurde jedoch eingestellt, da Rolipram im Vergleich zu herkömmlichen **Antidepressiva** keinen Zusatznutzen zeigen konnte.
ω_{unk} = 1.0 (imp)	Die klinische Entwicklung bei Depression wurde jedoch eingestellt, da Rolipram im Vergleich zu herkömmlichen **Antidepressiva** keinen Zusatznutzen zeigen konnte.
raw (neg)	Durch die Gabe von calcium- und magnesiumhaltigen Antacida nach oraler Überdosierung von Ofloxacin kann die Resorption infolge Bildung schwerlöslicher Komplexe verzögert werden.
raw (pos)	Durch die Gabe von calcium- und magnesiumhaltigen **Antacida** nach oraler Überdosierung von Ofloxacin kann die Resorption infolge Bildung schwerlöslicher Komplexe verzögert werden.
ω_{unk} = 0.01 (imp)	Durch die Gabe von **calcium-** und magnesiumhaltigen **Antacida** nach oraler Überdosierung von **Ofloxacin** kann die Resorption infolge Bildung schwerlöslicher Komplexe verzögert werden.
ω_{unk} = 0.2 (imp)	Durch die Gabe von **cal**cium- und magnesiumhaltigen **Antacida** nach oraler Überdosierung von **Oflo**xacin kann die Resorption infolge Bildung schwerlöslicher Komplexe verzögert werden.
ω_{unk} = 1.0 (imp)	Durch die Gabe von calcium- und magnesiumhaltigen **Antacida** nach oraler Überdosierung von Ofloxacin kann die Resorption infolge Bildung schwerlöslicher Komplexe verzögert werden.
raw (neg)	Bei einer **Überdosierung** von Fenetyllin werden große Mengen der **Neurotransmitter** Noradrenalin und Dopamin aus den Speichervesikeln im **zentralen Nervensystem** freigesetzt.
raw (pos)	Bei einer Überdosierung von Fenetyllin werden große Mengen der Neurotransmitter **Noradrenalin** und **Dopamin** aus den Speichervesikeln im zentralen Nervensystem freigesetzt.
ω_{unk} = 0.01 (imp)	Bei einer Überdosierung von **Fenetyllin** werden große Mengen der Neurotransmitter **Noradrenalin** und **Dopamin** aus den Speichervesikeln im zentralen Nervensystem freigesetzt.
ω_{unk} = 0.2 (imp)	Bei einer Überdosierung von **Fenetyllin** werden große Mengen der Neurotransmitter **Noradrenalin** und **Dopamin** aus den Speichervesikeln im zentralen Nervensystem freigesetzt.
ω_{unk} = 1.0 (imp)	Bei einer Überdosierung von Fenetyllin werden große Mengen der Neurotransmitter **Noradrenalin** und **Dopamin** aus den Speichervesikeln im zentralen Nervensystem freigesetzt.

Table 4: Original text samples (raw) and their annotation-imputed instances for certain ω_{unk} values. The text in **bold** denotes the annotated entities. The samples were chosen for illustration purposes. The annotation granularity reflects the token structure from the subword tokenizer of the GottBERT model.

Paragraph Retrieval for Enhanced Question Answering in Clinical Documents

Vojtěch Lanz and **Pavel Pecina**
Charles University, Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics
{lanz,pecina}@ufal.mff.cuni.cz

Abstract

Healthcare professionals often manually extract information from large clinical documents to address patient-related questions. The use of Natural Language Processing (NLP) techniques, particularly Question Answering (QA) models, is a promising direction for improving the efficiency of this process. However, document-level QA from large documents is often impractical or even infeasible (for model training and inference). In this work, we solve the document-level QA from clinical reports in a two-step approach: first, the entire report is split into segments and for a given question the most relevant segment is predicted by a NLP model; second, a QA model is applied to the question and the retrieved segment as context. We investigate the effectiveness of heading-based and naive paragraph segmentation approaches for various paragraph lengths on two subsets of the emrQA dataset (Pampari et al., 2018). Our experiments reveal that an average paragraph length used as a parameter for the segmentation has no significant effect on performance during the whole document-level QA process. That means experiments focusing on segmentation into shorter paragraphs perform similarly to those focusing on entire unsegmented reports. Surprisingly, naive uniform segmentation is sufficient even though it is not based on prior knowledge of the clinical document's characteristics.

1 Introduction

Healthcare professionals spend a lot of time going through extensive clinical documents, such as discharge summaries, to find specific answers to questions about their patients (Demner-Fushman et al., 2009). This process could be aided by Question Answering (QA) models, that search for substrings in the text of a clinical document to provide an evidence for a given question (Pampari et al., 2018).

Currently, encoder-based language models demonstrate strong performance in solving QA tasks (Lan et al., 2020; Zhang et al., 2020), even when we are looking for substrings in a multi-paragraph clinical context (Yue et al., 2020). However, the training process and inference of large language models (LLMs) on document-level QA require significant computational resources that are not always available. In addition, encoder-based and decoder-based models face difficulties in understanding and processing longer documents (Liu et al., 2023). A possible solution might be working with segments (paragraphs) of the document rather than full text.

To achieve this, we must first segment the document into paragraphs, then identify the relevant paragraph for a given question, and then apply an QA model only to the selected paragraph as the context instead of the full document text. This alone significantly facilitates healthcare professionals' work in finding answers in clinical texts, which is another reason why it is worth addressing the paragraph retrieval issue.

Clinical texts often lack structure (Richter-Pechanski et al., 2024; Gallego Donoso and Veredas, 2023) and contain information that is not expressed in natural language (Pampari et al., 2018). Moreover, each clinical text, authored by distinct doctors from various hospitals and even different countries, is arranged uniquely. Therefore, the task of segmenting a document into natural language paragraphs is inherently non-trivial. However, the question arises: is it necessary to segment clinical text into such structured paragraphs? Will a naive uniform segmentation without knowledge of the text itself have a similar performance?

Our work addresses QA on differently-sized paragraphs of clinical documents. First, given clinical document paragraphs and a given question, retrieve the most relevant paragraph. In the second step, we perform QA on that paragraph. In addi-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 580–590
August 16, 2024. ©2024 Association for Computational Linguistics

tion, we investigate the potential performance of the model on the QA task if the relevant paragraph is always predicted correctly.

We work with two subsets of the emrQA dataset: *Medication* and *Relations* (Pampari et al., 2018). We propose a heading-based segmentation into sections regarding different average paragraph lengths over all training clinical documents. We analyze the optimal average paragraph length to achieve the best performance and ensure that we preserve the context while keeping the paragraph as concise as possible. We then compare the performance of the encoding-based models on these segmentations with their performance on a naive segmentation approach. Finally, we demonstrate how LLMs perform under the same training conditions as encoder-based models. Our main contributions are the following:

- We demonstrate the feasibility of simplifying the document-level Question Answering (QA) challenge into a two-step task combining paragraph retrieval and paragraph-level QA.
- We propose a novel heading-based paragraph segmentation approach of emrQA data and compare its performance with naive segmentation.
- We present a comparative analysis of encoder-based and decoder-based models on the QA task, thus enriching the discussion on the optimal choice of architecture.

2 Related Work

The problem of question answering encompasses several different subtypes. One of them is to return an answer for a given question without any context (Berant et al., 2013). Another subtype involves text comprehension. For a given question and some context (such as a document or a paragraph), the task is to answer the question based on the content of the context, but the actual formulation of an answer is not restricted (Joshi et al., 2017). In our work, however, we focus on finding substrings in a given context that serve as both evidence and answer to a given question.

A significant resource in this field is the SQuAD dataset (Rajpurkar et al., 2016), containing questions, context paragraphs based on Wikipedia articles, and answer substrings. The dataset has been used to train and compare various neural methods, including encoder-based and decoder-based architectures (Lan et al., 2020; Zhang et al., 2020; Schmidt et al., 2024). This dataset was later extended into SQuAD v.2 (Rajpurkar et al., 2018), which also includes questions and corresponding paragraphs that do not contain an answer for a given question. As an alternative to this dataset in the clinical domain, the emrQA dataset (Pampari et al., 2018) was published. The emrQA dataset contains synthetically generated questions and substring answers for clinical reports from the n2c2 dataset (previously called i2b2). The emrQA consists of 5 subsets: *Medication*, *Relations*, *Heart disease*, *Obesity*, and *Smoking*, each focusing on different aspects and different complexity. From the emrQA dataset, the emrqa-msquad dataset (Eladio and Wu, 2024) was derived by summarizing clinical reports into single paragraphs as contexts and providing new manual annotated substring answers. However, this process removes the naturalness of clinical notes written by healthcare professionals. There is also the QA reading comprehension dataset in the medical scientific domain, which is BioASQ (Tsatsaronis et al., 2015). The dataset includes instances consisting of a question, ideal answer, PubMed medical article abstracts containing the answer, and the substring answers of all such related article abstracts.

In our work, we exploit the emrQA dataset (Pampari et al., 2018). Yue et al. (2020) analyzed the two largest subsets from the emrQA dataset in detail: *Medication* and *Relations*. They preprocessed and filtered these two subsets and trained encoder-based models, such as BERT-base (Devlin et al., 2018), BioBERT (Lee et al., 2019), and Clinical-BERT (Alsentzer et al., 2019), and then compared their performance. However, developments in the field have introduced other medically pre-trained encoder-based models, such as MedCPT (Jin et al., 2023), designed specifically for biomedical information retrieval, or BioLORD (Remy et al., 2024). Although the emrQA dataset authors perceive the analysis provided by Yue et al. (2020) as misleading due to the use of only 2 out of 5 subsets, for the purposes of our work, these two subsets with the same preprocessing and filtering are equally suitable. Therefore, our work indirectly follows up on the analysis conducted by Yue et al. (2020).

Another type of QA task involves multiple-choice questions. In the field of medicine, there is a PubMedQA dataset (Jin et al., 2019), which contains questions related to PubMed article abstracts. Furthermore, exam-like multiple-choice question

	Medication	Relations
Number of questions	222,957	904,590
Number of reports	262	426

Table 1: Basic statistics of both *Medication* and *Relations* subsets. Each question contains at least one answer present in the report.

datasets such as MedQA (Jin et al., 2020), MedMCQA (Pal et al., 2022), MMLU (Hendrycks et al., 2021) were published. These datasets have been used as benchmarks for LLMs, such as MediTron (Chen et al., 2023) and BioMistral-7B (Labrak et al., 2024), which are open-source LLMs pretrained on medical data. In addition to medical scientific and exam-like multiple-choice question datasets, Richter-Pechanski et al. (2024) focused on multiple-choice questions on German doctors' letters.

3 Setup

We solve the task of document-level QA on the emrQA dataset by a two-step method combining paragraph retrieval and paragraph-level QA. We analyze performance of the two tasks in combination and also separately. We follow the work of Yue et al. (2020) focusing on the *Medication* and *Relations* subsets only and applying the same data preprocessing. Table 1 shows the basic statistics of the two subsets. However, our results are not directly comparable due to the different random split into training, development, and test sets.

Throughout the rest of our study, we refer to these definitions:

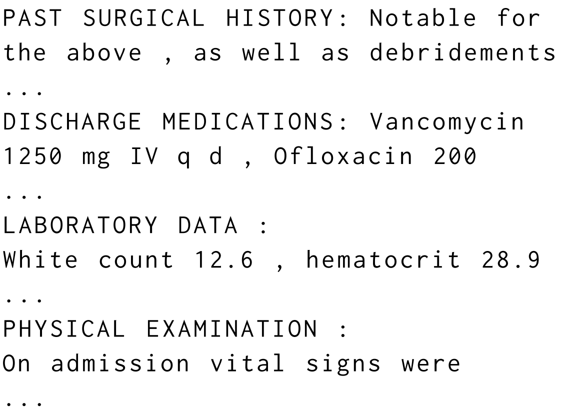
- **Paragraph Retrieval (PR)**: Given a question and n paragraphs (i.e. report segmented into n paragraphs) as input, the objective is to rank the paragraphs based on the confidence that they contain relevant information. The task is evaluated using precision at top 1 ($P@1$), precision at top 2 ($P@2$), and precision at top 3 ($P@3$) paragraphs. Ground truth relevant paragraphs are those containing an answer evidence to a given question defined in the emrQA dataset.

- **Oracle Paragraph-driven Question Answering (Oracle-QA)**: Given a question and an Oracle paragraph (guaranteed to contain the answer) the objective is to identify and extract a minimal substring from the paragraph that precisely addresses or answers the given question. The task is evaluated using the official SQuAD metrics (Rajpurkar et al., 2016), which are *F1* and *Exact Match* scores. We compare our predictions with the original form of the testing dataset generated by the filtration of Yue et al. (2020), i.e., with the dataset before the segmentation process.

- **Paragraph Retrieval–Question Answering (PR-QA)**: Given a question and n paragraphs (i.e. report segmented into n paragraphs), the goal is to identify and extract a substring from one of the paragraphs that precisely addresses or answers a given question. Evaluation of the task is based on the *F1* and *Exact Match* scores the same way as in the Oracle-QA task.

Yue et al. (2020) concluded that it is sufficient to use only 20% and 5% of training data to train models of the *Medication* and *Relations* subsets, respectively. Since a larger amount of training data has no effect, we use the same ratio of data samples for training. Their data instances consisted of triples of document+question+answer where the answer was guaranteed to be present in the document. Our data instances were generated as triples paragraph+question+answer where the answer was also guaranteed to be present in the paragraph and pairs paragraph+question where the corresponding answer was not present in the paragraph. For each question in a sampled training subset of a given report, we randomly select a paragraph from the same report that does not contain an answer. Thus, we have a balanced dataset where the number of paragraphs containing an answer matches the number of paragraphs without them.

In our experiments, we train the ClinicalBERT (Alsentzer et al., 2019) and BERT-base (Devlin et al., 2018) models, just as Yue et al. (2020) did. Additionally, we measure the performance of the MedCPT Article Encoder (Jin et al., 2023) model. Since we are working with a balanced dataset, it is necessary to specify how to handle cases where the answer is missing in the context. If there is no response in the dataset sample, the model is trained to predict the CLS token as a response prediction. During the inference, we apply the softmax function to the output logits and use its negative value as confidence that the paragraph contains the answer. Then, for a given question and segmented report into paragraphs, the model solves the QA problem for all paragraphs (we evaluate the Oracle-QA task using the ground truth relevant paragraph), ranks



```
PAST SURGICAL HISTORY: Notable for
the above , as well as debridements
...
DISCHARGE MEDICATIONS: Vancomycin
1250 mg IV q d , Ofloxacin 200
...
LABORATORY DATA :
White count 12.6 , hematocrit 28.9
...
PHYSICAL EXAMINATION :
On admission vital signs were
...
```

Figure 1: Example of report paragraph headings of both *Medication* and *Relations* subsets.

all results by confidence, and selects the substring of the paragraph with the highest confidence as the final answer prediction (and then the PR and PR-QA tasks are evaluated as well).

4 Document Segmentation

Our goal is to design a method for segmenting reports into natural language paragraphs that each contain all necessary context while minimizing unnecessary information. As Pampari et al. (2018) pointed out, the segmentation of clinical reports into sentences is not straightforward. These complications arise from factors such as the frequent use of dots in acronyms, list items, and values and the irregular alternation of uppercase and lowercase letters. Because our goal is to create concise paragraphs without losing context, we must ensure that paragraph boundaries do not disrupt sentence cohesion or, worse, do not appear in the middle of a word. Therefore, our initial step is to split each report into groups of complete sentences, ensuring that no sentence is fragmented across groups and that no substring of a response is split into two paragraphs.

To achieve this, we leverage the structure of the official emrQA dataset (Pampari et al., 2018). In this dataset, each report text is stored as a list of lines, with the answer evidence (in our case, the answer substring) being one of the report lines. Therefore, we set the condition that none of the sentence groups starts or ends in the middle of any line, ensuring that no answer substring is split into two paragraphs. We then split the *Medication* subset into groups of sentences if the following pattern for the end of a line is satisfied: a dot at the end of a line, preceded by five characters that are neither dots nor uppercase letters, and the next line starting with an uppercase letter (eventually this second line can also be an item of a numbered list, which means it could start with a number instead of an uppercase letter). Clinical notes in the subset of *Relations* are structured more clearly. Dots marking the end of a sentence are surrounded by spaces, while dots forming part of abbreviations are not. Thus, such space-surrounded dots at the end of a line indicate a sentence group boundary in the context of the *Relations* subset.

Another pattern we utilize as a criterion for segmentation contains a sequence of characters ending with a colon, indicating headings followed by corresponding content, as shown in Figure 1. Using the end of the previous line of such heading lines as a sentence group separator makes sense. To decrease the risk of detecting not-heading lines, we only consider uppercase titles for *Medication* when determining sentence group boundaries. In the case of the *Relations* subset, only lines ending with a colon preceded by space are considered, similar to the situation with dots.

Finally, we need to determine how we will group sentence groups together to create a final paragraph segmentation. By using the following regex pattern

```
(^([0-9]+[\s]*[\.\)][\s]*)?[A-Z][a-zA-Z\s\(\)]*:)
```

we identify all potential headings at the beginning of all sentence groups. Subsequently, we calculate how often these headings appear in the training data. We assume that frequently used titles signify sections generally discussed in clinical notes by healthcare professionals that do not need any additional context. Therefore, the question arises: what is the minimum number of occurrences of headings in the training data that we want to use for paragraph separations?

We call such segmentation as *heading-based segmentation*. As the range of possible headings serving as paragraph boundaries increases, the average length of paragraphs decreases. As shown in Table 2, segmenting reports using all detected headings yields PR-QA results comparable to those from unsegmented reports. Therefore, as part of our analysis, where we evaluate how frequently headings should be used as boundaries in segmentation, we assess our three tasks (PR, Oracle-QA, PR-QA) across different segmentations based on varying heading frequencies, resulting in different average segment lengths. This helps us understand

	Medication		Relations	
	F1	EM	F1	EM
MedCPT - *unsegmented reports*	68.33	27.48	94.69	87.68
MedCPT - *heading-based PR-QA*	64.79	26.63	94.05	88.44
BERT-base - *unsegmented reports*	70.09	30.07	95.04	89.32
BERT-base - *heading-based PR-QA*	68.19	30.23	95.15	91.28
ClinicalBERT - *unsegmented reports*	72.24	31.13	96.45	90.93
ClinicalBERT - *heading-based PR-QA*	70.80	31.19	96.44	92.69
Doc Reader (Yue et al. (2020))	*70.45*	*25.68*	*94.85*	*86.94*
Human-labled (Yue et al. (2020))	*74.70*	*26.0*	*95.40*	*92.00*

Table 2: Comparison of the results of pre-trained BERT models for QA applied to unsegmented reports and PR-QA applied to heading-based segmentations with the shortest possible average segment lengths. We also include the best results by Yue et al. (2020) evaluated on a test set sampled with a different random seed and their human-labeled analysis evaluated on a sampled subset of the test set.

the challenges involved in distinguishing relevant paragraphs from finding exact substring answers.

Despite the structured nature of the emrQA dataset, the rules for splitting the *Medication* and *Relations* subsets into paragraphs can be generalized to other clinical datasets with caution. Although different countries, hospitals, and doctors may structure their reports differently, there are often similar paragraphs and even common headings across various discharge summaries. This observation allows us to take the list of headings collected from the segmentation process of emrQA and use it when segmenting other discharge summaries. However, some level of preprocessing and postprocessing will always be necessary, as this method is not a one-size-fits-all solution for all clinical reports.

4.1 Medication

The newly created segmented datasets derived from the *Medication* subset need to be analyzed first. When segmenting reports into shorter paragraphs, more paragraph+question+answer triples are generated. This is because some questions have multiple possible answers in different document parts. By breaking the text into paragraphs, these question+answer pairs can be split into two or more. Figure 3 shows this expansion is minimal, only about $3-5\%$. However, this phenomenon does not affect the results since we compare our predictions with the original unsegmented reports. For questions with answers in multiple paragraphs, only answer, the most confident one, is selected for the evaluation. Figure 4 displays a list of 542 discovered headings sorted by their frequency of occurrence. We can see that the first third of the headings appear more frequently in all training reports. In contrast, two-thirds of the headings found do not appear to refer to traditional clinical sections. The average lengths of the segmented paragraphs are shown in Figure 5. Even though we collected headings only from training reports, it did not significantly impact the development and test sets. Anyway, it is still interesting to observe the wide range of segmented paragraph lengths.

We segmented the *Medication* subset into paragraphs with varying average lengths from hundreds to thousands of characters and evaluated the performance of ClinicalBERT (Alsentzer et al., 2019) model on all 3 tasks: PR, Oracle-QA, and PR-QA. We sampled the training dataset and trained the model with three different seeds. The results can be seen in the first row of Figure 2. Results are shown for segmentations with different average paragraph lengths corresponding to the x-axis.

The shorter the paragraphs, the easier the Oracle-QA task, but at the same time, the more paragraphs correspond to one report, making the PR task more challenging. Although the Oracle-QA task tends to perform better in both F1 and Exact Match scores for shorter paragraphs, the difference is not that significant. For an average paragraph length of 2500 characters and less, the model is not always confident in its top selection for the PR task. On the other hand, considering two or three top predictions, the correct paragraph is almost certainly included. After combining the predictions into the PR-QA chart, the resulting curve for the Exact Match remains constant for all possible average paragraph lengths. The curve of the F1 score is also constant, except for the shortest paragraphs. However, overall, the challenging part of the PR-QA is the Oracle-QA prediction.

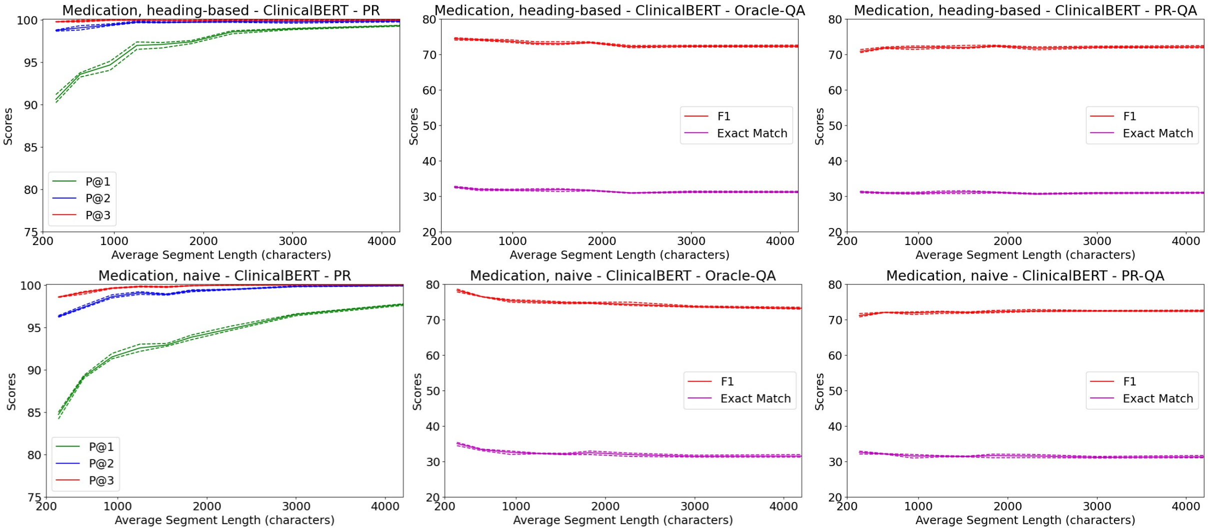


Figure 2: The comparison of heading-based and naive segmentation approaches for different average paragraph lengths using the ClinicalBERT (Alsentzer et al., 2019) model regarding all three tasks (PR, Oracle-QA, PR-QA) on the *Medication* subset. All values are computed as an average of three experiments based on different training seeds. The dashed lines visualize the range of score values.

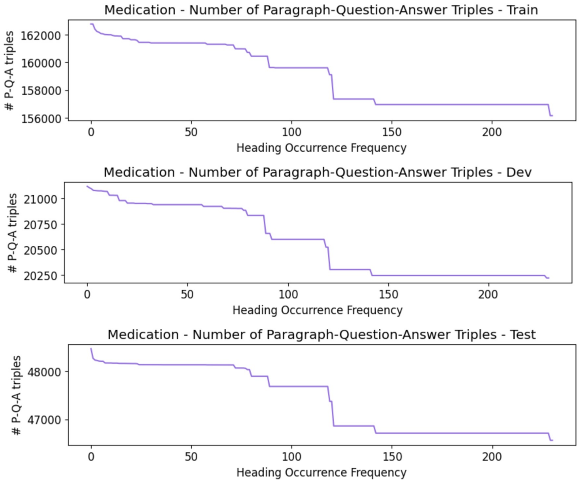


Figure 3: Number of paragraph+question+answer triples of the *Medication* subset in terms of the minimal occurrence frequency of headings we consider for segmentation for the training, development, and test sets.

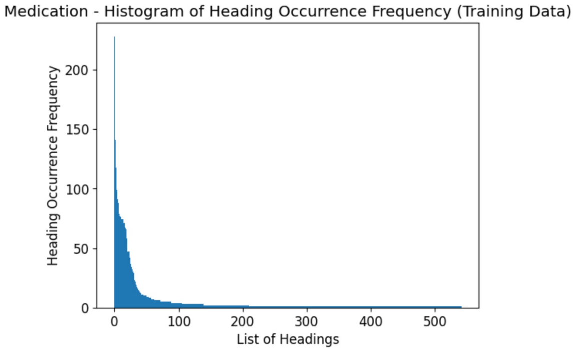


Figure 4: Histogram of heading occurrence frequencies for the *Medication* training set. The x-axis represents the heading positions within the sorted list of headings based on their occurrence frequency. Each point on the x-axis corresponds to a specific heading, with the y-value indicating its occurrence frequency. In total, there are 542 headings, with the most common appearing over 200 times. Many headings appear only once in the training data.

4.2 Relations

Although *Medication* and *Relations* are different subsets with different complexities, Figures 7, 8, and 9 indicate that the header-based segmentation approach behaves similarly for both. However, in this case, we found 953 headings, which we use for segmentation.

We conducted the same experiments on the *Relations* subset as we did in the case of the Medication section. The results, illustrated in the first row of Figure 6, cover the performance of the ClinicalBERT model (Alsentzer et al., 2019) on segmentations of *Relations* subset with varying average paragraph lengths. Given the lower complexity of the *Relations* subset compared to the *Medication*, the model performed better in all three tasks. The PR task achieved better than 98% of $P@1$, even for the shortest paragraphs. The Oracle-QA task indicates that the model performs notably better on shorter paragraphs so that PR-QA results could be improved. Following the combination, i.e., PR-QA task, a constant F1 curve was observed. Further-

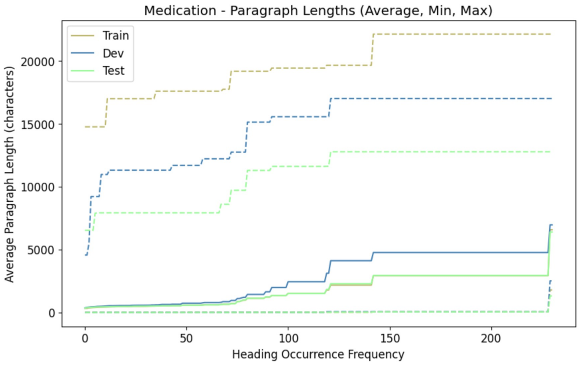


Figure 5: Average paragraph lengths regarding minimal occurrence frequency of headings we consider for *Medication* subset segmentation. The dashed lines show the minimum and maximum lengths of the segmented paragraphs.

more, there is a slight improvement in the Exact Match score by 1–2% when the shortest paragraphs are taken into account.

5 Is Segmentation into Paragraphs Necessary?

We have shown that heading-based paragraph segmentation has no significant effect on the PR-QA task except for improving the Exact Match score of the *Relations* when segmenting into shorter paragraphs. However, most clinical texts are unstructured and use unique text formatting; sometimes, finding a segmentation into coherent sections in the sense of meaning as well as syntax is not easy or even possible. To determine its necessity, we conduct experiments with *naive segmentation*.

We choose a target average segment length t to create the naive segmentation. Then, we calculate each report's length n and determine the rounded number of segments in the report as $p = \texttt{round}(n/t)$. Subsequently, we compute the actual average segment length of the report for the value of p as $r = n/p$. Finally, the report is divided into segments of r characters. Postprocessing is then applied across all segments and all answers in the report. In cases where an answer substring is part of two separate segments, we adjust the segment boundaries so that the entire answer is in one segment only.

The target average segment length t for naive segmentation is chosen to match the average segment lengths of the headings-based segmentation experiments. Specifically, for each measured segmentation level of the headings-based approach, we also measure the naive segmentation using a target average segment length equal to the average segment length of the given headings-based segmentation.

In Figures 2 and 6, we visualize the comparison between ClinicalBERT (Alsentzer et al., 2019) using heading-based and naive approaches. Except for segmentation with the shortest paragraphs, the choice of segmentation method has no noticeable effect on the PR-QA task. In the case of the shortest paragraphs of the *Relations* subset, the naive approach begins to decline in PR-QA performance, while the heading-based approach becomes more accurate. The reason for that is worse performance on the PR task as well as the Exact Match on the Oracle-QA. The performance of naive segmentation on the PR task is significantly worse. On the other hand, the Oracle-QA naive segmentation experiments show better results. The most confident segment contains less relevant content compared to heading-based segmentation, making it easier to find the correct substring as an answer (fewer relevant and potential words in the segment) if the segment itself is predicted correctly. Overall, the PR-QA performance of both heading-based and naive approaches is similar.

6 Paragraphs and LLMs

Considering the impact of segmentation into shorter paragraphs on the scores, it is noteworthy that it does not significantly affect them and may even enhance them. This observation suggests the potential for leveraging LLMs without the necessity for unlimited computational resources in future applications. In this study, we evaluate the performance of BioMistral-7B (Labrak et al., 2024) in the Oracle-QA task and compare it with MedCPT (Jin et al., 2023), BERT-base (Devlin et al., 2018), and ClinicalBERT (Alsentzer et al., 2019) models. BioMistral-7B (Labrak et al., 2024) is trained on question+paragraph+answer triplets where each paragraph contains an answer. Negative examples are omitted to focus solely on the Oracle-QA task. The model prompt is shown in Figure 10. For evaluation, the model's response is parsed into a JSON object, and the value of the "answer" field is extracted.

Table 3 presents the F1 and Exact Match results of the Oracle-QA task using heading-based segmentation with the shortest possible paragraphs, categorized into *Medication* and *Relations* subsets. The results demonstrate that BioMistral-7B

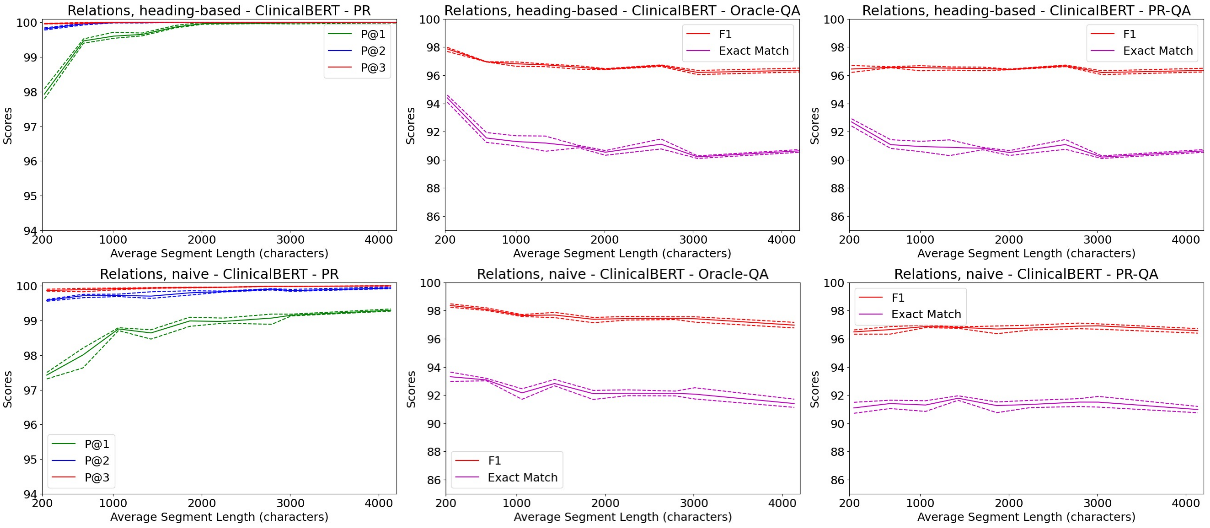


Figure 6: The comparison of heading-based and naive segmentation approaches for different average paragraph lengths using the ClinicalBERT (Alsentzer et al., 2019) model regarding all three tasks (PR, Oracle-QA, PR-QA) on the *Relations* subset. All values are computed as an average of three experiments based on different training seeds. The dashed lines visualize the range of score values.

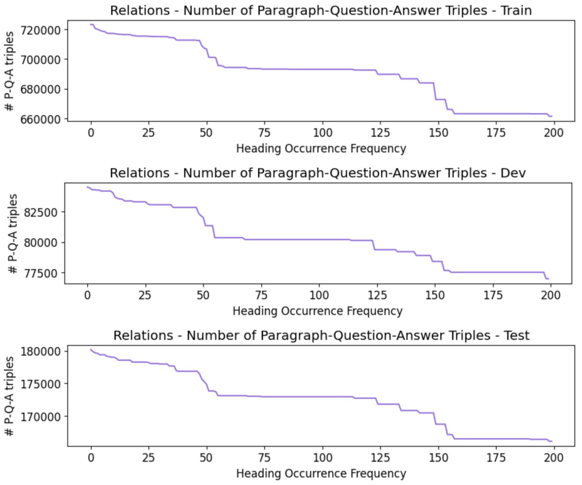


Figure 7: Number of paragraph+question+answer triples of the *Relations* subset in terms of the minimal frequency of occurrence of headings we consider for segmentation for the training, development, and test sets.

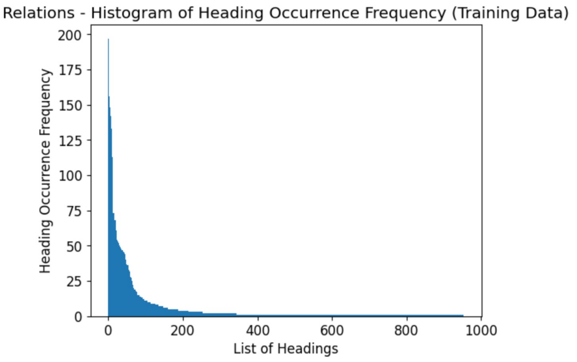


Figure 8: The histogram of heading occurrence frequencies for the *Relations* training set. The x-axis represents the heading positions within the sorted list of headings based on their occurrence frequency. Each point on the x-axis corresponds to a specific heading, with the y-value indicating its occurrence frequency. In total, there are 953 headings, with the most common appearing almost 200 times.

(Labrak et al., 2024) achieves competitive performance but still lags behind encoder-based models such as ClinicalBERT (Alsentzer et al., 2019) and BERT-base (Devlin et al., 2018). BioMistral-7B (Labrak et al., 2024) shows not only promising Exact Match scores compared to MedCPT (Jin et al., 2023), highlighting its potential in clinical QA tasks. However, further exploration is needed to optimize prompts and explore larger models to enhance performance.

7 Conclusions

Our study explores the efficiency of language models in addressing clinical document-level QA. We described an approach to perform heading-based segmentation and extract clinical report headings and found that segmenting documents into shorter sections through heading-based or naive approaches does not decline the performance of ClinicalBERT (Alsentzer et al., 2019), BERT-base (Devlin et al., 2018), or MedCPT (Jin et al., 2023) models. Paragraph length has no significant impact

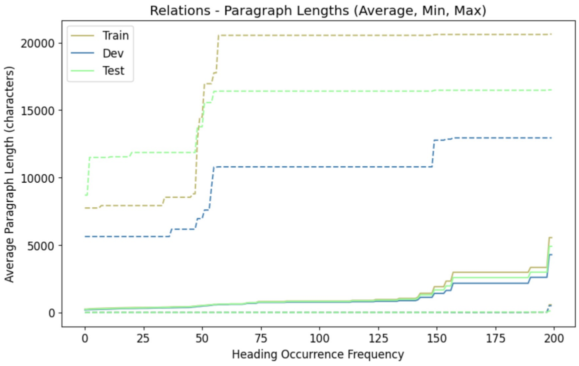
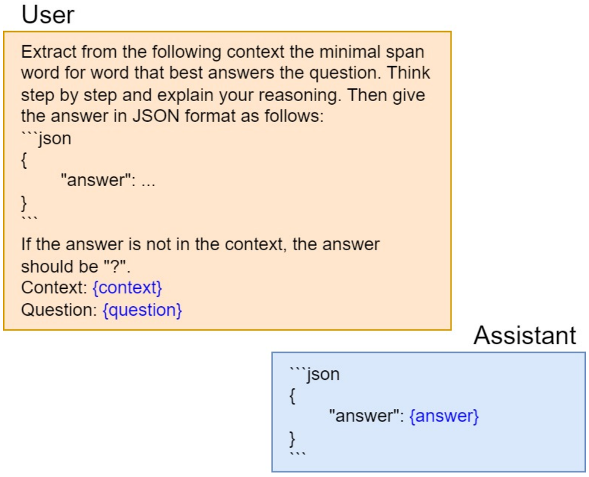


Figure 9: Average paragraph lengths regarding minimal occurrence frequency of headings we consider for *Relations* subset segmentation. The dashed lines show the minimum and maximum lengths of the segmented paragraphs.



Extract from the following context the minimal span word for word that best answers the question. Think step by step and explain your reasoning. Then give the answer in JSON format as follows:

~~~
```json
{
    "answer": ...
}
```
~~~

If the answer is not in the context, the answer should be "?".
Context: {context}
Question: {question}

~~~
```json
{
    "answer": {answer}
}
```
~~~

Figure 10: The prompt used for BioMistral-7B training and inference in the Oracle-QA task for extracting answers from a context given a particular question.

on the QA task. Furthermore, knowledge of clinical document characteristics is unnecessary since naive segmentation performs similarly to heading-based segmentation. The main difference is that naive segmentation is more challenging for paragraph retrieval but easier for question answering. In both cases, however, we observe that the correct segment containing the answer is almost always found within the three most confident paragraph retrieval predictions.

Leveraging LLMs like BioMistral-7B (Labrak et al., 2024) shows potential for document-level clinical QA tasks even when computational resources are limited. However, there is still room for improvement and it is necessary to explore other pre-trained LLMs with different training approaches. It remains an open question how the segmented paragraph approach would affect results and behavior on more complex tasks or datasets. Further research is needed to evaluate these methods in more challenging QA scenarios to fully understand their impact and potential.

	m-F1	**m-EM**	**r-F1**	**r-EM**
MedCPT	70.7	28.3	96.7	91.5
BERT-base	73.0	31.9	97.5	94.0
ClinicalBERT	74.4	32.5	97.9	94.4
BioMistral	66.6	29.8	94.4	89.0

Table 3: F1 (**F1**) and Exact Match (**EM**) Oracle-QA results using the heading-based segmentation of the shortest possible paragraphs for both *Medication* (**m**) and *Relations* (**r**) subsets.

Acknowledgments

This work has received support and funding from the European Union's Horizon Europe research and innovation programme project *RES-Q plus* (Grant Agreement No. 101057603).

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA. Association for Computational Linguistics.

Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, Alexandre Sallinen, Alireza Sakhaeirad, Vinitra Swamy, Igor Krawczuk, Deniz Bayazit, Axel Marmet, Syrielle Montariol, Mary-Anne Hartley, Martin Jaggi, and Antoine Bosselut. 2023. Meditron-70b: Scaling medical pretraining for large language models. *Preprint*, arXiv:2311.16079.

Dina Demner-Fushman, Wendy W. Chapman, and Clement J. McDonald. 2009. What can natural language processing do for clinical decision support? *Journal of Biomedical Informatics*, 42(5):760–772. Biomedical Natural Language Processing.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Jimenez Eladio and Hao Wu. 2024. emrqa-msquad: A medical dataset structured with the squad v2.0 framework, enriched with emrqa medical information. *Preprint*, arXiv:2404.12050.

Fernando Gallego Donoso and Francisco Veredas. 2023. Icb-uma at biocreative viii @ amia 2023 task 2 symptemist (symptom text mining shared task). In *Proceedings of the BioCreative VIII Challenge and Workshop: Curation and Evaluation in the era of Generative Models*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Preprint*, arXiv:2009.03300.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2020. What disease does this patient have? A large-scale open domain question answering dataset from medical exams. *CoRR*, abs/2009.13081.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

Qiao Jin, Won Kim, Qingyu Chen, Donald C Comeau, Lana Yeganova, W John Wilbur, and Zhiyong Lu. 2023. Medcpt: Contrastive pre-trained transformers with large-scale pubmed search logs for zero-shot biomedical information retrieval. *Bioinformatics*, 39(11):btad651.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. Biomistral: A collection of open-source pretrained large language models for medical domains. *Preprint*, arXiv:2402.10373.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. *Preprint*, arXiv:1909.11942.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *CoRR*, abs/1901.08746.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts. *Preprint*, arXiv:2307.03172.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pages 248–260. PMLR.

Anusri Pampari, Preethi Raghavan, Jennifer J. Liang, and Jian Peng. 2018. emrqa: A large corpus for question answering on electronic medical records. *CoRR*, abs/1809.00732.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

François Remy, Kris Demuynck, and Thomas Demeester. 2024. BioLORD-2023: semantic textual representations fusing large language models and clinical knowledge graph insights. *Journal of the American Medical Informatics Association*, page ocae029.

Phillip Richter-Pechanski, Philipp Wiesenbach, Dominic M. Schwab, Christina Kiriakou, Nicolas Geis, Christoph Dieterich, and Anette Frank. 2024. Clinical information extraction for low-resource languages with few-shot learning using pre-trained language models and prompting. *Preprint*, arXiv:2403.13369.

Maximilian Schmidt, Andrea Bartezzaghi, and Ngoc Thang Vu. 2024. Prompting-based synthetic data generation for few-shot question answering. *Preprint*, arXiv:2405.09335.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael Alvers, Dirk Weißenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, Yannis Almirantis, John Pavlopoulos, Nicolas Baskiotis, Patrick Gallinari, Thierry Artieres, Axel-Cyrille Ngonga Ngomo, Norman Heino, Eric Gaussier, Liliana Barrio-Alvers, and Georgios

Paliouras. 2015. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC Bioinformatics*, 16:138.

Xiang Yue, Bernal Jimenez Gutierrez, and Huan Sun. 2020. Clinical reading comprehension: A thorough analysis of the emrQA dataset. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4474–4486, Online. Association for Computational Linguistics.

Zhuosheng Zhang, Junjie Yang, and Hai Zhao. 2020. Retrospective reader for machine reading comprehension. *Preprint*, arXiv:2001.09694.

CID at RRG24: Attempting in a Conditionally Initiated Decoding of Radiology Report Generation with Clinical Entities

Yuxiang Liao*, Yuanbang Liang*, Yipeng Qin, Hantao Liu, Irena Spasić
School of Computer Science and Informatics, Cardiff University, UK
{liaoy11, liangy32, qiny16, liuh35, spasici}@cardiff.ac.uk

Abstract

Radiology Report Generation (RRG) seeks to leverage deep learning techniques to automate the reporting process of radiologists. Current methods are typically modelling RRG as an image-to-text generation task that takes X-ray images as input and generates textual reports describing the corresponding clinical observations. However, the wording of the same clinical observation could have been influenced by the expression preference of radiologists. Nevertheless, such variability can be mitigated by normalizing textual reports into structured representations such as a graph structure. In this study, we attempt a novel paradigm for incorporating graph structural data into the RRG model. Our approach involves predicting graph labels based on visual features and subsequently initiating the decoding process through a template injection conditioned on the predicted labels. We trained and evaluated our model on the BioNLP 2024 Shared Task on Large-Scale Radiology Report Generation and submitted our results to the ViLMedic RRG leaderboard. Although our model showed a moderate ranking on the leaderboard, the results provide preliminary evidence for the feasibility of this new paradigm, warranting further exploration and refinement.

1 Introduction

Radiology Report Generation (RRG) seeks to liberate radiologists from the repetitive reporting process, allowing them to focus on revising the reports and thereby enhancing the accuracy and efficiency of clinical communication. As a multi-modality task, RRG models usually employ the encoder-decoder architecture, where the encoder is a vision model that is responsible for extracting visual features from radiology images while the decoder is a language model that is responsible for converting visual features into narrative reports. Compared with the general image captioning task, the clinical observations in radiology images are more subtle. Moreover, the wording of the same clinical observation could have been influenced by the expression preference of radiologists. This raises a challenge to the model's learning ability in terms of extracting fine-grained visual features and generate accurate clinical narratives.

Our recent review of this field proposed that structured reports can alleviate the inherent diversity of natural language, thus contributing to more accurate results in the model training and evaluation (Liao et al., 2023). Benefiting from the advent of RadGraph (Jain et al., 2021), a graph-based representation of clinically significant fine-grained information extracted from reports, recent research has commenced utilising such structured representation of reports to enhance the RRG models. Relevant studies can be broadly classified into two paradigms. One paradigm fuses the graph features with visual features, letting the decoder learn how to generate the next word from a given input and the fused features (Wang et al., 2022; Yan et al., 2023; Yang et al., 2022; Li et al., 2023). Another paradigm focuses on graph generation based on the visual features and decouples the visual features from the decoding stage, allowing the language model to learn solely how to generate text based on the predicted graph (Nooralahzadeh et al., 2021; Xiong et al., 2024).

This has sparked our interest, as it raises a research question of whether there exists a new paradigm that can explicitly leverage graph structures to improve the quality of generative language models, while also enabling visual features to supplement the predicted graph with missing information. Based on this idea, we attempt a novel approach, whereby the predicted graphs are fed into a template prompt, replacing the traditional special token as the initial input to the decoder, aiming to enable a clearer query to the associated

[*] Contributed equally.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 591–596
August 16, 2024. ©2024 Association for Computational Linguistics

image features during the generation process.

2 Related Work

In early research on RRG, many studies introduced disease labels to enhance their models (Jing et al., 2018; Yin et al., 2019; Yuan et al., 2019; Harzig et al., 2019; Wang et al., 2018). As research progressed, some studies began to explore the use of graph to replace disease labels as it can represent more fine-grained information (Zhang et al., 2020; Li et al., 2019). The graph is considered as a normalized representation of a report in terms of the the key information entities and their relationships (Jain et al., 2021).

To utilise graph data, Nooralahzadeh et al. (2021) and Yan et al. (2023) proposed modelling RRG as a pipeline of image-to-graph and graph-to-text tasks. Xiong et al. (2024) followed the same paradigm although their study contained only the first part. In contrast, Wang et al. (2022) interpreted the RRG as an image-to-text task where a graph prediction module was appended to the visual encoder. Additionally, the graph features were combined with visual features and passed to the text decoder, allowing the text decoder to learn to attend to different features. Yang et al. (2022) and Li et al. (2023) employed a similar feature fusion approach, yet their graph was not directly predicted from visual features, but rather retrieved from the paired report of a similar image identified by comparing their visual features.

3 Method

3.1 Vision Encoder Decoder Model

Our model comprises a pre-trained Transformer-based vision model as the encoder and a pre-trained language model as the decoder. A cross-attention layer and a language model head are appended to the decoder to support generation.

Let I denote a radiology image and T denote the corresponding report text. A cross-attention feature $\mathbf{\Phi}_{T,I}$ is computed by Attention($\mathbf{Q}, \mathbf{K}, \mathbf{V}$), where the query $\mathbf{Q}$ represents the encoded text features $\mathbf{\Phi}_T$, and the key $\mathbf{K}$ and value $\mathbf{V}$ represent the encoded visual features $\mathbf{\Phi}_I$. During the training stage, $\mathbf{\Phi}_{T,I}$ is passed to a language model head to generate a complete text sequence $\hat{T}$ at once. The model is updated by the cross-entropy loss between the probability of the predicted tokens in $\hat{T}$ and the target tokens in T. During the inference stage, the model takes an image as the encoder input and a special token as the decoder input and generates the next token through an auto-regressive decoding process.

In this architecture, the prevailing methods that combine the graph or label features with visual features can be interpreted as providing more information to K and V to be queried. However, we assume that the visual features have sufficient information, thus, we aim to enhance Q to better utilise the information from the visual features.

3.2 Graph Label Selection

We first customized a structured reporting tool based on RadGraph to preprocess the raw text. RadGraph is an information extraction tool that can convert narrative radiology reports into graphs. In RadGraph, each node is an entity that corresponds to a continuous span of text. Each edge is a uni-directional relation that connects two entities. Entities are assorted into four types: Anatomy, Observation-Present/Absent/Uncertain. Relations are assorted into three types: Suggestive-Of, Located-At, and Modify. We refer the reader to the original paper for details (Jain et al., 2021). We refined RadGraph by combining the Observation and Anatomy nodes that are linked with a Located-At edge such as "lung hyperinflate", while the other nodes were omitted. Label's text content was lemmatized. We selected labels that have appeared in more than 5,000 reports, resulting in 79, 22 and 10 label classes representing present, absent, and uncertain, respectively. For any other label, we assigned a dummy label to represent the corresponding category. Therefore, each report can be enhanced by 114 informative labels.

3.3 Multi-label Classification

Let the L^{ctg} denote the labels of a specific category $ctg = \{present, absent, uncertain\}$ extracted from a report T. We first introduced an auxiliary task of multi-label classification (MLC) between the encoding-decoding process:

$$\mathbf{p}^{ctg} = \sigma(\mathrm{FFNN}(\theta^{ctg}; \overline{\mathbf{\Phi}_I})), \quad (1)$$

where FFNN represents a feed-forward neural network classifier with learning parameters θ that predicts the probability distribution $\mathbf{p}^{ctg}$ of labels in a specific category ctg, taking the average pooled visual features $\overline{\mathbf{\Phi}_I}$ as input to get optimised θ^{ctg}. The classification loss is computed by the cross-entropy loss between the predicted probabilities and the target labels for all categories.

By incorporating the MLC task into the model, the overall objective is thus to optimize the text generation loss and label classification loss, denoted as $\mathcal{L}_{all} = \lambda_T \mathcal{L}_T + \lambda_{MLC} \mathcal{L}_{MLC}$, where λ_T and λ_{MLC} are pre-defined weights that balance two losses.

3.4 Conditionally Initiated Decoding

To enhance the query Q in the cross-attention layer, we inject the labels directly into the decoder as its initial input. Specifically, the labels are rewritten as a label text sequence via a template: "Observation present: []; absent: []; uncertain: []. Describe them in detail: ". Each label string is filled into one of the brackets according to its category while the dummy label is filled as "others".

During training, we employ a teacher-forcing approach that uses the target labels as the source labels to fill the template. Therefore, the decoder input sequence is formed as "<BOS>label text sequence<EOS><EOS>report text sequence<EOS>". During the inference stage, we combine the predicted labels from the three classifiers and select no more than top-k labels with probabilities exceeding the threshold as the source label for the template. Therefore, the initial decoder input is transformed into "<BOS>label text sequence<EOS><EOS>" and the next token is generated through an autoregressive decoding process. A workflow of our model is illustrated in Figue 1.

3.5 Batch Inference

When performing batch inference on the data, the inconsistency in the number and length of the activated label poses an alignment issue when constructing the input tensor. To address this, we employ left padding during the inference stage to ensure the generated tokens and the initial decoder input are semantically continuous. Furthermore, the padding tokens are also marked out from the decoder attention mask to prevent them from influencing other tokens.

4 Experiments

4.1 Experimental Settings

Our experiments are conducted on the BioNLP 2024 Shared Task on Large-Scale Radiology Report Generation (Xu et al., 2024), which proposes the first standard to the community regarding the use of the dataset and evaluation metrics.

4.1.1 Datasets

This shared task provides the first large-scale collection of RRG datasets based on MIMIC-CXR (Johnson et al., 2019), CheXpert (Chambon et al., 2024), OpenI (Demner-Fushman et al., 2015), PadChest (Bustos et al., 2020) and CANDID-PTX (Vayá et al., 2020). Each data item represents a radiology examination consisting of at least one X-ray image and two pieces of text corresponding to the findings and impression sections of the radiology report. Any non-English reports were translated into English via GPT-4. The provided dataset has been split into training, validation, testing subsets. Testing data were further split into public and hidden subsets.

4.1.2 Metrics

The models are automatically evaluated by the ViLMedic metric package (Delbrouck et al., 2022b) using the following metrics: Bertscore (Zhang et al., 2019), Bilingual Evaluation Understudy: 4-gram (BLUE-4) (Papineni et al., 2002), Recall-Oriented Understudy for Gisting Evaluation: Longest Common Subsequence (ROUGE-L) (Lin, 2004), F1-RadGraph: partial (Delbrouck et al., 2022a) and all-micro-F1-CheXbert (Smit et al., 2020).

4.1.3 Implementation Details

Our model uses Swinv2-base (Liu et al., 2022) as the visual encoder and Roberta-base (Liu et al., 2019) as the text decoder. The encoder takes only the first image as input for each data. The decoder input sequence accepts a maximum of 512 tokens, where any surplus tokens are truncated. The decoder input sequences are padded to the longest sequence in each batch. We trained the model on the finding and impression respectively. In all experiments, the model was trained on NVIDIA RTX 4090 24G for 30 epochs using a learning rate of 1e-4 and a batch size of 12. A weight decay of 0.01 is set to the encoder and decoder. We updated the model with the AdamW optimizer using a linear scheduler with a warmup ratio of 0.1, and a gradient clipping set to 1. λ_T and λ_{MLC} are set to 1 and 5, respectively. During inference, we adopt the beam search strategy and set the beam size to 3 and the maximum generation length to 128. For the conditionally initiated decoding, we selected no more than 10 labels with probabilities exceeding 0.5.

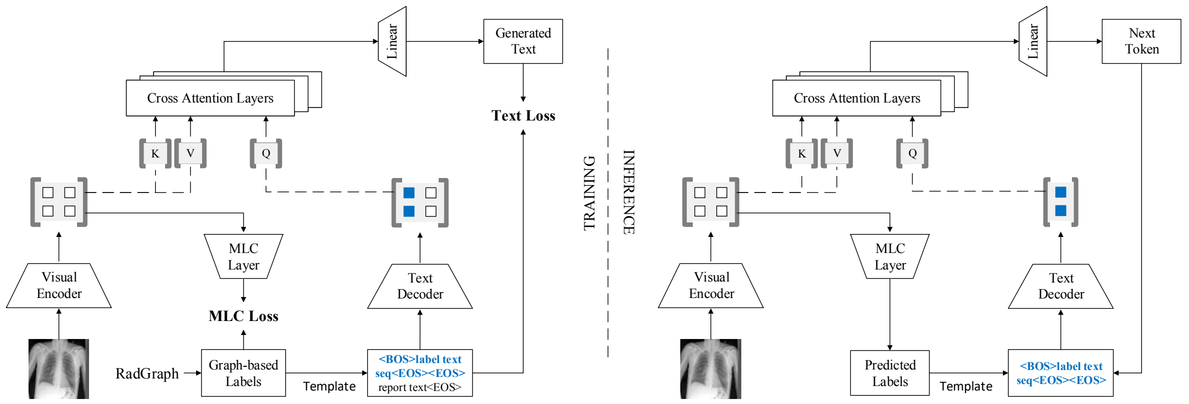


Figure 1: The workflow of our model during training (left) and inference (right). The blue text represents the conditionally initiated decoder input, which substitutes the original special token that functioned as the decoder's first input.

4.2 Results and Discussion

The performance of our model in generating the findings and impression sections of a report are illustrated in Table 1. The performance gap between the findings and impression sections is mainly due to the early termination of training to meet the system submission deadline. Although our model only exhibits a medium result on the ViLMedic RRG leaderboard (Delbrouck et al., 2022b), we assume this prototype model is feasible and has the potential to be improved.

Table 1: Model performance on the public and hidden test subsets.

Data subsets	BLEU4	ROUGEL	Bertscore	F1-cheXbert	F1-RadGraph
Public test-set					
Findings	8.29	24.38	52.28	51.13	22.26
Impression	5.25	18.71	41.72	42.86	15.13
Hidden test-set					
Findings	7.46	23.3	50.89	50.47	21.45
Impression	7.13	20.41	43.67	39.64	15.19

Firstly, we utilised only the first image from each data item as the encoder input. Given that a radiologist may refer to multiple images when composing a report, using the image features extracted from a single image may result in information loss when multiple images are available. However, the number of available images for each data item is uncertain, raising a challenge to the visual model in terms of its adaption.

Secondly, properly utilising the graph data remains unexplored yet has direct impacts on various aspects of the model. For example, the selection of graph labels can directly affect the learning difficulty of multi-label classification (MLC). If the number of labels is too small, the amount of information provided to the Conditionally Initiated Decoding (CID) may be limited even with good MLC performance. Conversely, if the number of labels is too large, the MLC performance may be significantly affected, making it impossible to provide accurate information to the CID during inference. Currently, our MLC on the finding section achieved precision/recall of 76%/40%, 63%/39% and 31%/15% on the present, absent, and uncertain labels, respectively. The trade-off between these factors requires further study. Besides, the impact of the label text template on the decoder remains unclear.

Thirdly, the current selection of model hyperparameters and the base pre-trained models for the encoder and decoder was based on experience. Due to time constraints, we did not systematically explore other combinations. Comprehensive experiments with the hyperparameters and the pre-trained models are also required in future work.

5 Conclusion

In this study, we propose a novel approach for utilizing graph structural data to support RRG. This approach involves predicting graph labels based on visual features and leveraging the predicted labels to initialize the decoder input through a template injection. We evaluated our model following the BioNLP 2024 Shared Task 1: Radiology Report Generation, where the results have been submitted to the ViLMedic RRG leaderboard. We discuss the limitations of our preliminary RRG model and the initial experiments and outline several directions for improving our model. Our model and codes are available on GitHub (Liao, 2024).

6 Limitations

Firstly, our model only accepts a single radiology image per data item as input, whereas the data item could contain multiple images, resulting in the loss of significant input information. Secondly, it remains uncertain to what extent the quality of the generated text is influenced by the decoder input initialized with graph-structured data. Thirdly, the selection of current hyperparameters and pretrained models is based on intuition rather than appropriate experimentation. More details have been discussed in Section 4.2.

Finally, our model requires an additional GPU-CPU-GPU switch during inference, leading to increased time costs. Specifically, the Conditionally Initiated Decoding process requires switching to the CPU to dynamically construct the decoder input with a tokenizer for each batch. However, we suppose that this issue can be addressed by pretokenizing and caching the template text and all graph labels. The improvement the model efficiency will be conducted in our future work.

References

Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria De La Iglesia-Vaya. 2020. Padchest: A large chest x-ray image dataset with multi-label annotated reports. *Medical image analysis*, 66:101797.

Pierre Chambon, Jean-Benoit Delbrouck, Thomas Sounack, Shih-Cheng Huang, Zhihong Chen, Maya Varma, Steven QH Truong, Chu The Chuong, and Curtis P. Langlotz. 2024. Chexpert plus: Augmenting a large chest x-ray dataset with text radiology reports, patient demographics and additional image formats. *Preprint*, arXiv:2405.19538.

Jean-Benoit Delbrouck, Pierre Chambon, Christian Bluethgen, Emily Tsai, Omar Almusa, and Curtis Langlotz. 2022a. Improving the factual correctness of radiology report generation with semantic rewards. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4348–4360.

Jean-benoit Delbrouck, Khaled Saab, Maya Varma, Sabri Eyuboglu, Pierre Chambon, Jared Dunnmon, Juan Zambrano, Akshay Chaudhari, and Curtis Langlotz. 2022b. Vilmedic: a framework for research at the intersection of vision and language in medical ai. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 23–34.

Dina Demner-Fushman, Marc D. Kohli, Marc B. Rosenman, Sonya E. Shooshan, Laritza Rodriguez, Sameer Antani, George R. Thoma, and Clement J. McDonald. 2015. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association*, 23(2):304–310.

Philipp Harzig, Yan-Ying Chen, Francine Chen, and Rainer Lienhart. 2019. Addressing data bias problems for chest x-ray image report generation. In *British Machine Vision Conference*.

Saahil Jain, Ashwin Agrawal, Adriel Saporta, Steven Truong, Du Nguyen Duong Nguyen Duong, Tan Bui, Pierre Chambon, Yuhao Zhang, Matthew Lungren, Andrew Ng, Curtis Langlotz, Pranav Rajpurkar, and Pranav Rajpurkar. 2021. Radgraph: Extracting clinical entities and relations from radiology reports. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Baoyu Jing, Pengtao Xie, and Eric Xing. 2018. On the automatic generation of medical imaging reports. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2577–2586, Melbourne, Australia. Association for Computational Linguistics.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317.

Christy Y. Li, Xiaodan Liang, Zhiting Hu, and Eric P. Xing. 2019. Knowledge-driven encode, retrieve, paraphrase for medical image report generation. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press.

Mingjie Li, Bingqian Lin, Zicong Chen, Haokun Lin, Xiaodan Liang, and Xiaojun Chang. 2023. Dynamic graph enhanced contrastive learning for chest x-ray report generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3334–3343.

Yuxiang Liao. 2024. Code for cid at rrg24: Attempting in a conditionally initiated decoding of radiology report generation with clinical entities. Github. Accessed on: March. 03, 2024.

Yuxiang Liao, Hantao Liu, and Irena Spasić. 2023. Deep learning approaches to automatic radiology report generation: A systematic review. *Informatics in Medicine Unlocked*, 39:101273.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019.

RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv e-prints*.

Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. 2022. Swin transformer v2: Scaling up capacity and resolution. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11999–12009.

Farhad Nooralahzadeh, Nicolas Perez Gonzalez, Thomas Frauenfelder, Koji Fujimoto, and Michael Krauthammer. 2021. Progressive transformer-based generation of radiology reports. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2824–2832, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Y Ng, and Matthew Lungren. 2020. Combining automatic labelers and expert annotations for accurate radiology report labeling using bert. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1500–1519.

Maria De La Iglesia Vayá, Jose Manuel Saborit, Joaquim Angel Montell, Antonio Pertusa, Aurelia Bustos, Miguel Cazorla, Joaquin Galant, Xavier Barber, Domingo Orozco-Beltrán, Francisco García-García, et al. 2020. Bimcv covid-19+: a large annotated dataset of rx and ct images from covid-19 patients. *arXiv preprint arXiv:2006.01174*.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, and Ronald M. Summers. 2018. Tienet: Text-image embedding network for common thorax disease classification and reporting in chest x-rays. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9049–9058.

Zhanyu Wang, Mingkang Tang, Lei Wang, Xiu Li, and Luping Zhou. 2022. A medical semantic-assisted transformer for radiographic report generation. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*, pages 655–664, Cham. Springer Nature Switzerland.

Yiheng Xiong, Jingsong Liu, Kamilia Zaripova, Sahand Sharifzadeh, Matthias Keicher, and Nassir Navab. 2024. Prior-radgraphformer: Prior-knowledge-enhanced transformer for generating radiology graphs from x-rays. In *Graphs in Biomedical Image Analysis, and Overlapped Cell on Tissue Dataset for Histopathology: 5th MICCAI Workshop, GRAIL 2023 and 1st MICCAI Challenge, OCELOT 2023, Held in Conjunction with MICCAI 2023, Vancouver, BC, Canada, September 23, and October 4, 2023, Proceedings*, page 54–63, Berlin, Heidelberg. Springer-Verlag.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and “discharge me!”. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Sixing Yan, William K. Cheung, Keith Chiu, Terence M. Tong, Ka Chun Cheung, and Simon See. 2023. Attributed abnormality graph embedding for clinically accurate x-ray report generation. *IEEE Transactions on Medical Imaging*, 42(8):2211–2222.

Shuxin Yang, Xian Wu, Shen Ge, S. Kevin Zhou, and Li Xiao. 2022. Knowledge matters: Chest radiology report generation with general and specific knowledge. *Medical Image Analysis*, 80:102510.

Changchang Yin, Buyue Qian, Jishang Wei, Xiaoyu Li, Xianli Zhang, Yang Li, and Qinghua Zheng. 2019. Automatic generation of medical imaging diagnostic report with hierarchical recurrent neural network. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 728–737.

Jianbo Yuan, Haofu Liao, Rui Luo, and Jiebo Luo. 2019. Automatic radiology report generation based on multi-view image fusion and medical concept enrichment. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, pages 721–729, Cham. Springer International Publishing.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Yixiao Zhang, Xiaosong Wang, Ziyue Xu, Qihang Yu, Alan Loddon Yuille, and Daguang Xu. 2020. When radiology report generation meets knowledge graph. In *AAAI Conference on Artificial Intelligence*, volume 34, Palo Alto, California USA. AAAI Press.

MAIRA at RRG24: A specialised large multimodal model for radiology report generation

Shaury Srivastav[1], Mercy Ranjit[1], Fernando Pérez-García[2], Kenza Bouzid[2], Shruthi Bannur[2], Daniel C. Castro[2], Anton Schwaighofer [2], Harshita Sharma[2], Maximilian Ilse[2], Valentina Salvatelli[2], Sam Bond-Taylor[2], Fabian Falck[2], Anja Thieme[2], Hannah Richardson[2], Matthew P. Lungren[3], Stephanie L. Hyland[2], Javier Alvarez-Valle[2]

[1]Microsoft Research India,
[2]Health Futures, Microsoft Research,
[3]Microsoft Health and Life Sciences

Correspondence: meranjit@microsoft.com

Abstract

This paper discusses the participation of the MSR MAIRA team in the Large-Scale Radiology Report Generation Shared Task Challenge, as part of the BioNLP workshop at ACL 2024. We present a radiology-specific multimodal model designed to generate radiological reports from chest X-Rays (CXRs). Our proposed model combines a CXR-specific image encoder RAD-DINO (Pérez-García et al., 2024) with a Large Language Model (LLM) based on Vicuna-7B, via a multi-layer perceptron (MLP) adapter. Both the adapter and the LLM have been fine-tuned in a single-stage training setup to generate radiology reports. Experimental results indicate that a joint training setup with findings and impression sections improves findings prediction. Additionally, incorporating lateral images alongside frontal images when available further enhances all metrics. More information and resources about MAIRA can be found on the project website: `http://aka.ms/maira`.

1 Introduction

An impactful application of natural language generation in the medical field involves the creation of support systems that interpret patient X-ray images and produce a draft report detailing the clinical findings in these images. Such systems have the potential to enhance and expedite radiology reporting workflows. In this regard, a Shared Task Challenge for Radiology Report Generation has been organised as part of the ACL 2024 BioNLP workshop[1] (RRG24; Xu et al., 2024). This shared task challenge uses the first large-scale collection of radiology report generation datasets based on MIMIC-CXR (Johnson et al., 2019), CheXpert (Chambon et al., 2024), PadChest (Bustos et al., 2020), BIMCV-COVID19 (Vayá et al., 2020), and Open-i (Nguyen et al., 2022) datasets. This paper covers the participation of the MAIRA (Multimodal AI for Radiology Applications) team at Microsoft Research in this challenge.

We build on the architecture and training approach from our earlier work MAIRA-1 (Hyland et al., 2024). This approach combines a CXR-specific image encoder (RAD-DINO, Pérez-García et al. (2024)) with a pretrained LLM (Vicuna-7B 1.5, Chiang et al. (2023)) via an adapter which is an MLP of 4 layers. Both LLM and adapter are finetuned in a single stage for the task of radiology report generation, while the image encoder is pretrained following the self-supervised DINOv2 approach (Oquab et al., 2024). For this competition, we produce a variant of MAIRA-1 which is trained only on public data, and further extended it to incorporate lateral images. We share the following outcomes:

1. A joint training setup for findings and impression prediction improves the metrics for findings generation.

2. Coupling lateral views with frontal views when available shows improvement in both the clinical and lexical metrics.

3. We show that scaling the model size from Vicuna-7B to 13B further helps to improve all the metrics. We also show that smaller models like Phi-3-mini with 3.8B parameters is on-par with the larger models.

[1] `https://stanford-aimi.github.io/RRG24/`

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 597–602
August 16, 2024. ©2024 Association for Computational Linguistics

Table 1: Number of studies with a given section across the data sources and splits in the RRG24 challenge dataset.

Split	Section	MIMIC-CXR	CheXpert	Open-i	PadChest	BIMCV-COVID19
Train	Findings	148 374	45 491	3252	101 752	45 525
	Impression	181 166	181 619	3628	–	–
Validation	Findings	3799	1112	85	2641	1202
	Impression	4650	4589	92	–	–

2 Method

2.1 Dataset

The dataset statistics of the RRG24 challenge are available in Table 1. Each study can have multiple frontal and/or lateral images. We processed this dataset into two versions: frontal-only (Table 2), where each record contains only one frontal image, and frontal-with-laterals (Table 3), where each record contains one frontal or a frontal and a lateral image. Hence, each record in the RRG24 dataset may be split into more than one record in these datasets.

Table 2: RRG24 frontal-only dataset. Each record corresponds to exactly one frontal image.

Section	Train	Test	Val.	Hidden
Findings	359 351	2900	9215	1133
Impression	381 075	3205	9691	1495
Total	740 426	6105	18 906	2628

Table 3: RRG24 frontal-with-laterals dataset. Each record corresponds to a frontal image (F) with or without a lateral image (L).

Section	View	Train	Test	Val.	Hidden
Findings	F	174 977	1193	4420	562
	F+L	196 023	1897	5081	629
Impression	F	226 673	1392	5792	825
	F+L	165 835	2005	4193	734
Total		763 508	6487	19 486	2750

The image encoder is trained using the images from the MIMIC-CXR (Johnson et al., 2019), CheXpert (Chambon et al., 2024), PadChest (Bustos et al., 2020), NIH-CXR (Wang et al., 2017), and BRAX (Reis et al., 2022) datasets. Both frontal and lateral view images were used. The dataset statistics by source are available in Table 4. The images in the validation and test sets of RRG24 challenge were excluded.

Table 4: Training datasets for the image encoder.

Source	View	No. of images
MIMIC-CXR	Frontal, lateral	367 932
CheXpert	Frontal, lateral	218 180
PadChest	Frontal, lateral	156 432
NIH-CXR	Frontal	112 120
BRAX	Frontal, lateral	41 260

2.2 Model architecture

We leverage the MAIRA-1 (Hyland et al., 2024) architecture that consists of a CXR-specific image encoder, an adapter layer and an LLM. The image encoder (Pérez-García et al., 2024) is a ViT-B model (Dosovitskiy et al., 2020). The input image resolution is 518×518. The LLM is Vicuna-7B 1.5 (Chiang et al., 2023). The adapter is an MLP with 4 layers with a hidden size of 1024 for all the layers. The prompt setup we used is available in Table 5.

2.3 Training details

For the image encoder training, we follow RAD-DINO (Pérez-García et al., 2024) and use an image-level objective, a patch-level objective, and a regulariser to encourage uniform span of the features within a batch. We initialised the model with the weights of the pre-trained DINOv2 ViT-B (Oquab et al., 2024) and trained with the chest X-ray images in Table 4 for an additional 60k training steps, with an effective batch size of 640. We used the AdamW optimizer with a base learning rate of 0.001 and a cosine learning rate scheduler with linear warm-up.

For MAIRA-RRG24 training, we keep the image encoder frozen. We just train the adapter and the LLM with a standard auto-regressive language modelling loss. We use a cosine learning rate scheduler with a warm-up of 0.03 and learning rate of

Table 5: Prompt for the different tasks. F: frontal, L: lateral.

Setting	View	Prompt
Findings	F	Given the frontal image `{image_tokens}`, provide a description of the findings in the radiology study.
	F+L	Given the frontal image `{image_tokens}` and the lateral image `{lateral_image_tokens}`, provide a description of the findings in the radiology study.
Impression	F	Given the frontal image `{image_tokens}`, provide a summary impression in the radiology study.
	F+L	Given the frontal image `{image_tokens}` and the lateral image `{lateral_image_tokens}`, provide a summary impression in the radiology study.

Table 6: Experimental settings. F: frontal, L: lateral.

Setting	View	Task
Findings (F)	F	findings prediction
Findings + Impression (F)	F	findings and impression prediction (multi-task)
Findings (F+L)	F+L	findings prediction
Findings + Impression (F+L)	F+L	findings and impression prediction (multi-task)

2×10^{-5}. We train for 3 epochs with a global batch size of 128. During evaluation, as there could be multiple predictions for a study (a study could have more than one frontal/lateral images), we use GPT-4 with the prompt defined in Table 10 to select the best prediction.

2.4 Evaluation metrics

We use the vilmedic package (Delbrouck et al., 2022b) for computing the metrics. ROUGE-L (Lin, 2004), BLEU-4 (Papineni et al., 2002) and BERTScore (Zhang et al., 2019) were used to measure the lexical performance. F1-CheXbert (Smit et al., 2020) and F1-Radgraph (Delbrouck et al., 2022a) were used to measure the clinical performance.

3 Experiments

We perform experiments in four settings as defined in Table 6: single-task training for findings generation, joint(multitask) training for findings and impression generation, and combinations of both with or without lateral images alongside frontal images. We use dataset versions in Table 2 and Table 3 when we train for frontal only setup and frontal and lateral setup respectively. We call the model trained with the multitask setting for findings and impression prediction using the frontal and lateral images as MAIRA-RRG24. We also trained MAIRA-RRG24 with a smaller LLM, Phi-3-mini-4k-instruct (Abdin et al., 2024) with 3.8B parameters. We also performed an additional scaling experiment for the findings generation task with laterals (third setting in Table 6) using the Vicuna 7B and 13B versions.

3.1 Results

The results of the experiments are available in Table 7. We find that a joint training setup involving findings and impression prediction tasks shows a slight improvement in the findings prediction metrics compared to training for findings prediction alone. Additionally, training with lateral images in addition to frontal images further improves all the metrics. The best experimental setup, which is the Findings+Impression (F+L) setting involves joint training for findings and impression prediction tasks, along with the inclusion of lateral images when available. The results of our best setting on the hidden test set are presented in Table 9. We also trained our best setting, with a smaller model Phi-3-mini-4k-instruct and got better or competitive results in all the metrics. The results of the model scaling experiment in Table 8 demonstrate that a larger model size helps to improve the metrics.

Table 7: MAIRA-RRG24 – Experimental results for findings generation task on the public-test set.

Setting	BLEU-4	ROUGE-L	BERTScore	F1-CheXbert	F1-RadGraph
Findings (F)	10.61	26.70	54.54	52.64	24.57
Findings+Impression (F)	10.88	26.86	54.50	55.55	24.68
Findings (F+L)	11.20	26.59	54.53	56.95	24.84
Findings+Impression (F+L)	12.26	28.00	55.76	59.71	26.33
Findings+Impression (F+L) (Phi-3-mini-4k-instruct)	14.84	29.17	58.91	55.87	27.07

Table 8: MAIRA-RRG24 – Model scaling experiment. Public test results for Findings (F+L) setting.

LLM	BLEU-4	ROUGE-L	BERTScore	F1-CheXbert	F1-RadGraph
vicuna-7b-v1.5	11.20	26.59	54.53	56.95	24.84
vicuna-13b-v1.5	12.17	27.86	55.62	59.66	26.21

Table 9: MAIRA-RRG24 – Hidden test set results for the Findings+Impression (F+L) setting.

Task	BLEU-4	ROUGE-L	BERTScore	F1-CheXbert	F1-RadGraph
Findings Generation	11.24	26.58	54.22	57.87	25.48
Impression Prediction	11.66	28.48	51.62	53.27	25.26

Table 10: GPT-4 prompt for selecting the best report for a study when there are multiple records.

```
You are an AI assistant who helps to select the best
radiology report from multiple reports written for
the same patient. User will send you a list of
reports. You will select the best report based on
the below criteria.
1. It has the best complete list of findings that
contains the findings from other reports as well.
2. Do not contain hallucinations like comparison
to a previous report and other noisy details.
3. The writing style matches closely with that of
a radiologist.
Return just the number of the index of the list
corresponding to the best report. The index starts
with 0.
```

4 Limitations

MAIRA-RRG24 does not have access to the prior studies and hence it may generate ‘hallucinated’ references to prior studies (Bannur et al., 2023).

5 Conclusion

We have presented MAIRA-RRG24, a radiology-adapted large multimodal model based on the MAIRA-1 architecture (Hyland et al., 2024) with a RAD-DINO-like (Pérez-García et al., 2024) image encoder, trained exclusively with the data available for the RRG24 challenge (Xu et al., 2024). It exhibits competitive performance in both lexical and clinical metrics. It benefits from a domain-specific image encoder, a joint training setup for findings and impression prediction leveraging lateral images when available.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Qin Cai, Martin Cai, Caio César Teodoro Mendes, Weizhu Chen, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Yen-Chun Chen, Yi-Ling Chen, Parul Chopra, Xiyang Dai, Allie Del Giorno, Gustavo de Rosa, Matthew Dixon, Ronen Eldan, Victor Fragoso, Dan Iter, Mei Gao, Min Gao, Jianfeng Gao, Amit Garg, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Jamie Huynh, Mojan Javaheripi, Xin Jin, Piero Kauffmann, Nikos Karampatziakis, Dongwoo Kim, Mahoud Khademi, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Ce Liu, Mengchen Liu, Weishung Liu, Eric Lin, Zeqi Lin, Chong Luo, Piyush Madan, Matt Mazzola, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Xin Wang, Lijuan Wang, Chunyu Wang, Yu Wang, Rachel Ward, Guanhua Wang, Philipp Witte, Haiping Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Sonali Yadav, Fan Yang, Jianwei Yang, Ziyi Yang, Yifan Yang, Donghan Yu, Lu Yuan, Chengruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

Shruthi Bannur, Stephanie Hyland, Qianchu Liu, Fernando Pérez-García, Maximilian Ilse, Daniel C. Castro, Benedikt Boecking, Harshita Sharma, Kenza Bouzid, Anja Thieme, Anton Schwaighofer, Maria Wetscherek, Matthew P. Lungren, Aditya Nori, Javier Alvarez-Valle, and Ozan Oktay. 2023. Learning to exploit temporal structure for biomedical vision-language processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15016–15027.

Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria De La Iglesia-Vaya. 2020. PadChest: A large chest x-ray image dataset with multi-label annotated reports. *Medical image analysis*, 66:101797.

Pierre Chambon, Jean-Benoit Delbrouck, Thomas Sounack, Shih-Cheng Huang, Zhihong Chen, Maya Varma, Steven QH Truong, Chu The Chuong, and Curtis P. Langlotz. 2024. Chexpert plus: Augmenting a large chest x-ray dataset with text radiology reports, patient demographics and additional image formats. *Preprint*, arXiv:2405.19538.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality.

Jean-Benoit Delbrouck, Pierre Chambon, Christian Bluethgen, Emily Tsai, Omar Almusa, and Curtis Langlotz. 2022a. Improving the factual correctness of radiology report generation with semantic rewards. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4348–4360.

Jean-benoit Delbrouck, Khaled Saab, Maya Varma, Sabri Eyuboglu, Pierre Chambon, Jared Dunnmon, Juan Zambrano, Akshay Chaudhari, and Curtis Langlotz. 2022b. ViLMedic: a framework for research at the intersection of vision and language in medical AI. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 23–34.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Stephanie L. Hyland, Shruthi Bannur, Kenza Bouzid, Daniel C. Castro, Mercy Ranjit, Anton Schwaighofer, Fernando Pérez-García, Valentina Salvatelli, Shaury Srivastav, Anja Thieme, Noel Codella, Matthew P. Lungren, Maria Teodora Wetscherek, Ozan Oktay, and Javier Alvarez-Valle. 2024. MAIRA-1: A specialised large multimodal model for radiology report generation. *Preprint*, arXiv:2311.13668.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. MIMIC-CXR, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Ha Q. Nguyen, Khanh Lam, Linh T. Le, Hieu H. Pham, Dat Q. Tran, Dung B. Nguyen, Dung D. Le, Chi M. Pham, Hang T. T. Tong, Diep H. Dinh, Cuong D. Do, Luu T. Doan, Cuong N. Nguyen, Binh T. Nguyen, Que V. Nguyen, Au D. Hoang, Hien N. Phan, Anh T. Nguyen, Phuong H. Ho, Dat T. Ngo, Nghia T. Nguyen, Nhan T. Nguyen, Minh Dao, and Van Vu. 2022. VinDr-CXR: An open dataset of chest X-rays with radiologist's annotations. *Scientific Data*, 9(1):429.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. 2024. DINOv2: Learning robust visual features without supervision. *Preprint*, arXiv:2304.07193.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Fernando Pérez-García, Harshita Sharma, Sam Bond-Taylor, Kenza Bouzid, Valentina Salvatelli, Maximilian Ilse, Shruthi Bannur, Daniel C Castro, Anton Schwaighofer, Matthew P Lungren, et al. 2024. RAD-DINO: Exploring scalable medical image encoders beyond text supervision. *Preprint*, arXiv:2401.10815.

Eduardo P. Reis, Joselisa P. Q. de Paiva, Maria C. B. da Silva, Guilherme A. S. Ribeiro, Victor F. Paiva, Lucas Bulgarelli, Henrique M. H. Lee, Paulo V. Santos, Vanessa M. Brito, Lucas T. W. Amaral, Gabriel L.

Beraldo, Jorge N. Haidar Filho, Gustavo B. S. Teles, Gilberto Szarf, Tom Pollard, Alistair E. W. Johnson, Leo A. Celi, and Edson Amaro. 2022. BRAX, Brazilian labeled chest X-ray dataset. *Scientific Data*, 9(1):487.

Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Y Ng, and Matthew Lungren. 2020. Combining automatic labelers and expert annotations for accurate radiology report labeling using bert. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1500–1519.

Maria De La Iglesia Vayá, Jose Manuel Saborit, Joaquim Angel Montell, Antonio Pertusa, Aurelia Bustos, Miguel Cazorla, Joaquin Galant, Xavier Barber, Domingo Orozco-Beltrán, Francisco García-García, et al. 2020. BIMCV COVID-19+: a large annotated dataset of RX and CT images from COVID-19 patients. *Preprint*, arXiv:2006.01174.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M. Summers. 2017. ChestX-Ray8: Hospital-scale chest X-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

AIRI at RRG24: LLaVa with specialised encoder and decoder

V. Samokhin[1], **M. Munkhoeva**[1], **D. Umerenkov**[1]*, **E. Kuzmina**[1,2], **I. Oseledets**[1,2], **D. Dylov**[1,2]
[1]AIRI, [2] Skoltech
{samokhin,munkhoeva,umerenkov,kuzmina,oseledets,dylov}@airi.net

Abstract

We present a new approach to generating the "Findings" and "Impression" sections in the chest X-rays radiology reports, developed as part of the shared radiology task at BioNLP 2024. By integrating a DINOv2 vision encoder trained on medical data with specialized biomedical large language model using the LLaVA framework, our method addresses complex medical semantics and diverse findings in imaging. We use datasets from PadChest, BIMCV-COVID19, CheXpert, OpenI, and MIMIC-CXR. The evaluation metrics demonstrate our method's effectiveness and the potential for automating the generation of radiology reports.

1 Introduction

The automatic generation of radiology reports from chest X-rays is a challenging and significant task in the field of biomedical natural language processing (BioNLP). The growing volume of medical imaging data and the limited number of radiologists necessitate the development of robust automated systems to assist in report generation. Such systems not only have the potential to improve clinical workflow efficiency but also to ensure consistency and comprehensiveness in radiological interpretations.

In recent years, advancements in deep learning and natural language processing have paved the way for innovative approaches to tackle this task. The new approaches typically involve the integration of convolutional neural networks (CNNs) or visual transformers for image feature extraction with recurrent neural networks (RNNs) or transformers for text generation (Selivanov et al., 2023). Despite the progress, challenges such as capturing complex medical semantics, handling diverse imaging findings, and ensuring the clinical accuracy of generated reports remain.

This paper explores a new method for generating the Findings and Impression sections of radiology reports from chest X-rays. Our approach is to combine a vision encoder, self-trained on medical data, with specialized biomedical LLM for text generation, using LLaVA framework. This work was done as a part of Radiology Report Generation ahared task at BioNLP 2024 Workshop (Xu et al., 2024) using the data provided by the organizers. The metrics were calculated using the ViLMedic platform (Delbrouck et al., 2022b).

2 Data

2.1 Training and validation data

The data from 5 datasets where combined to create the competition training and validation datasets: PadChest(Bustos et al., 2020), BIMCV-COVID19(Vayá et al., 2020), CheXpert(Chambon et al., 2024), OpenI(Demner-Fushman et al., 2012) and MIMIC-CXR(Johnson et al., 2019). The training and validation sets are grouped by study but not by subjects. The official language of PadChest and BIMCV-COVID19 is Spanish, and their reports have been translated using GPT-4 by the shared task organizers.

The data consists of radiology studies, each containing one or more chest X-ray images in various projections. Each study also includes Impression and Finding texts. Some studies have only the Impression or only the Findings section, while others have both.

2.2 Testing Data

The studies in the test sets are unseen studies provided by organizers. Public test sets for impression and findings contain both study images and ground truth texts while private test set contains only images.

[*] V.S., M.M., and D.U. contributed equally.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 603–607
August 16, 2024. ©2024 Association for Computational Linguistics

Dataset	**Findings**	**Impressions**
PadChest	101,752	-
BIMCV-COVID19	45,525	-
CheXpert	45,491	181,619
OpenI	3,252	3,628
MIMIC-CXR	148,374	181,166
Total	**344,394**	**366,413**

Table 1: Training dataset statistics.

Dataset	**Findings**	**Impressions**
CheXpert	1,112	4,589
BIMCV-COVID19	1,202	-
PadChest	2,641	-
OpenI	85	92
MIMIC-CXR	3,799	4,650
Total	**8,839**	**9,331**

Table 2: Validation dataset statistics.

Dataset	**Findings**	**Impressions**
public test-set	2,692	2,967
hidden test-set	1,063	1,428

Table 3: Testing datasets statistics

2.3 Data preprocessing

Due to technical limitations, we only used the first two images from each study. Studies with only one image were not further processed. For studies with more than one image, the first two images were stitched together horizontally. No additional preprocessing was applied to the texts.

3 Evaluation metrics

In the evaluation of radiology report summarization systems, several metrics are commonly used to assess the performance and accuracy of the generated summaries. These metrics ensure that the summaries produced by the models are not only syntactically and semantically correct but also factually accurate. The metrics used in this competition where BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), BERTScore (Zhang et al., 2019), F1-CheXbert (Smit et al., 2020), and F1-RadGraph (Delbrouck et al., 2022a).

3.1 BLEU (Bilingual Evaluation Understudy)

BLEU-4: This metric is widely used for evaluating machine translation systems. It measures the precision of n-grams in the generated summary by comparing it to one or more reference summaries. BLEU-4 specifically considers 4-gram overlaps, providing a robust measure of how many 4-grams in the generated text appear in the reference texts. However, it does not account for recall or the contextual meaning of words.

3.2 ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

ROUGE-L: ROUGE is predominantly used for evaluating automatic text summarization. ROUGE-L measures the longest common subsequence (LCS) between the generated summary and the reference summary. This metric emphasizes recall by capturing the longest sequence of words that appear in both the generated and reference summaries, thus reflecting the ability of the summary to include important information.

3.3 BERTScore

BERTScore: This metric computes the similarity between the generated and reference texts using pre-trained BERT embeddings. It calculates a similarity score for each token in the candidate sentence with each token in the reference sentence. BERTScore accounts for the semantic meaning of words, making it more robust against synonyms and paraphrasing compared to BLEU and ROUGE.

3.4 F1-CheXbert

F1-CheXbert: This metric evaluates the factual correctness of the generated summaries with a focus on specific medical conditions mentioned in radiology reports. CheXbert is a specialized tool designed to extract medical observations from radiology reports. The F1 score is calculated based on the precision and recall of these extracted observations, ensuring that the generated summaries accurately reflect the medical conditions described in the reference summaries.

3.5 F1-RadGraph

F1-RadGraph: Similar to F1-CheXbert, this metric evaluates the factual correctness of the summaries using the RadGraph dataset. RadGraph focuses on extracting entities and the relations between them from radiology reports. The F1-RadGraph score measures the accuracy of these extractions, comparing the generated summaries to the reference summaries to ensure that the critical entities and their relationships are accurately captured.

These metrics collectively provide a comprehensive evaluation framework for radiology report summarization systems. BLEU and ROUGE focus on the surface-level n-gram overlaps, while BERTScore provides a deeper semantic evaluation. F1-CheXbert and F1-RadGraph ensure the factual accuracy of medical details, which is crucial for clinical applications.

4 Methods and Results

We used the LLaVA model (Liu et al., 2024) with a DINOv2 encoder (Oquab et al., 2023) and OpenBio-LLM-8B (Ankit Pal, 2024) as a text decoder. The whole pipeline was implemented using HuggingFace's *transformers* (Wolf et al., 2020) and *trl* (von Werra et al., 2020) libraries.

For image encoding we used a DINOv2 Model with the following parameters:

- **Model**: ViT-base 14, initialized from torch.hub's `dinov2_vitb14`
- **Patch size**: 14
- **Number of parameters**: 86M
- **Time and Resources:** 4xA100 80GB GPUs, Training Total Time: 2 days
- **Dataset:** MIMIC-CXR Train, downsampled to 518 px
- **Batch size per GPU: 50**
- **Base Learning Rate:** 0.001

For text generation, we used OpenBioLLM-8B, an open-source language model designed specifically for the biomedical domain.

- **Training type:** LoRA on LLM's Attention matrices (r=64, alpha=16) + MM projector
- **Architecture:** OpenBio-LLM-8B + in-house DINOv2 trained on MIMIC-CXR Train
- **Time and Resources:** 5 epochs, 8xH100 80GB GPUs, DeepSpeed Zero-3; Training Total Time: 2 days
- **Batch size per GPU:** 8, gradient accumulation: 2
- **Base Learning Rate:** 0.001, cosine schedule, warmup: 0.15
- **Optimizer:** Adam

Vanilla approach to fine-tune LLaVA model with language model unfreezed resulted in rapid overfitting, thus we opted for PEFT methods (Mangrulkar et al., 2022), namely LoRA (Hu et al., 2022).

We used the same model for generating both impression and findings, using different prompts: either ”Write findings for this X-ray.” or ”Write impression for this X-ray.”.

We used the following system prompt, inspired by LLaVA-Med (Li et al., 2024):”You are a large language and vision assistant. You are designed to assist human with a variety of medical visual content and clinical research tasks using natural language. Follow the instructions carefully and provide clinically valid answers.”

Our results on hidden test sets are presented in Table 4 and Table 5.

Table 4: Findings - hidden test set (1063 samples)

Metric	e-health csiro	maira	airi
BLEU4	11.68	11.24	9.97
ROUGEL	26.16	26.58	25.82
Bertscore	53.80	54.22	52.42
F1-cheXbert	57.49	57.87	54.25
F1-RadGraph	28.67	25.48	25.29

Table 5: Impressions - hidden test set (1428 samples)

Metric	e-health csiro	maira	airi
BLEU4	12.33	11.66	10.91
ROUGEL	28.32	28.48	27.46
Bertscore	50.94	51.62	49.55
F1-cheXbert	56.97	53.27	52.32
F1-RadGraph	27.83	25.26	24.67

Our relatively simple model demonstrates strong performance in generating radiology reports. We attribute this success to the use of a specialized image encoder and a specialized large language model. Future improvements can be realized by employing larger models and fully using the available image data, which would likely enhance the competition metrics of the generated reports.

References

Malaikannan Sankarasubbu Ankit Pal. 2024. Openbiollms: Advancing open-source large language models for healthcare and life sciences. https://huggingface.co/aaditya/OpenBioLLM-Llama3-70B.

Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria De La Iglesia-Vaya. 2020. Padchest: A large chest x-ray image dataset with multi-label annotated reports. *Medical image analysis*, 66:101797.

Pierre Chambon, Jean-Benoit Delbrouck, Thomas Sounack, Shih-Cheng Huang, Zhihong Chen, Maya Varma, Steven QH Truong, Chu The Chuong, and Curtis P. Langlotz. 2024. Chexpert plus: Augmenting a large chest x-ray dataset with text radiology reports, patient demographics and additional image formats.

Jean-Benoit Delbrouck, Pierre Chambon, Christian Bluethgen, Emily Tsai, Omar Almusa, and Curtis Langlotz. 2022a. Improving the factual correctness of radiology report generation with semantic rewards. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4348–4360.

Jean-benoit Delbrouck, Khaled Saab, Maya Varma, Sabri Eyuboglu, Pierre Chambon, Jared Dunnmon, Juan Zambrano, Akshay Chaudhari, and Curtis Langlotz. 2022b. Vilmedic: a framework for research at the intersection of vision and language in medical ai. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 23–34.

Dina Demner-Fushman, Sameer Antani, Matthew Simpson, and George R Thoma. 2012. Design and development of a multimodal biomedical information retrieval system. *Journal of Computing Science and Engineering*, 6(2):168–177.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. 2024. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems*, 36.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Alexander Selivanov, Oleg Y Rogov, Daniil Chesakov, Artem Shelmanov, Irina Fedulova, and Dmitry V Dylov. 2023. Medical image captioning via generative pretrained transformers. *Scientific Reports*, 13(1):4171.

Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Y Ng, and Matthew Lungren. 2020. Combining automatic labelers and expert annotations for accurate radiology report labeling using bert. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1500–1519.

Maria De La Iglesia Vayá, Jose Manuel Saborit, Joaquim Angel Montell, Antonio Pertusa, Aurelia Bustos, Miguel Cazorla, Joaquin Galant, Xavier Barber, Domingo Orozco-Beltrán, Francisco García-García, et al. 2020. Bimcv covid-19+: a large annotated dataset of rx and ct images from covid-19 patients. *arXiv preprint arXiv:2006.01174*.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, and Shengyi Huang. 2020. Trl: Transformer reinforcement learning. `https://github.com/huggingface/trl`.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October. Association for Computational Linguistics.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and “discharge me!”. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand, August. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

iHealth-Chile-1 at RRG24: In-context Learning and Finetuning of a Large Multimodal Model for Radiology Report Generation

Diego Campanini [2], **Oscar Loch** [1,2,3], **Pablo Messina** [1,2,3],
Rafael Elberg [1,2,3], **and Denis Parra** [1,2,3]
[1] Department of Computer Science, Pontifical Catholic University of Chile.
[2] Millennium Institute for Intelligent Healthcare Engineering (iHEALTH), Chile.
[3] National Center for Artificial Intelligence (CENIA), Chile.
{oscar.loch,pamessina,rafael.elberg}@uc.cl,
dparra@ing.puc.cl, diego.campanini@ing.uchile.cl

Abstract

This paper presents the approach of the iHealth-Chile-1 team for the shared task of Large-Scale Radiology Report Generation at the BioNLP workshop, inspired by progress in large multimodal models for processing images and text. In this work, we leverage LLaVA, a Visual-Language Model (VLM), composed of a vision-encoder, a vision-language connector or adapter, and a large language model able to process text and visual embeddings. We achieve our best result by enriching the input prompt of LLaVA with the text output of a simpler report generation model. With this enriched-prompt technique, we improve our results in 4 of 5 metrics (BLEU-4, Rouge-L, BertScore, and F1-RadGraph,), only doing in-context learning. Moreover, we provide details about different architecture settings, fine-tuning strategies, and dataset configurations. Models parameters can be found in HuggingFace [1].

1 Introduction

The task of radiology report generation (RRG) from medical imaging through deep neural networks is an active area of research (Monshi et al., 2020; Messina et al., 2022). For one thing, addressing and solving this task can help radiologists in identifying anomalies from one or more input images, as well as save them time on administrative chores like typing text reports. Thus, doctors can spend more time with patients rather than clinical software (Topol, 2019). There have been several methods introduced in recent years to address this task but only recently the progress in open-source multimodal generative systems has opened the room for improving performance by integrating different modalities (text and images) in the same model. In this article, we describe our work leveraging the multimodal model LLaVA (Liu et al., 2023) to address this task.

There are several options to leverage LlaVA for this challenge, such as utilizing the original version LLaVA-1.0 (Liu et al., 2023), the clinically finetuned version LLaVA-Med (Li et al., 2023), as well as the newest version LLaVA-1.5 (Liu et al., 2024). Due to hardware limitations, in this challenge, we used the language model component with 7 billion parameters (LLaMA 1.0 and Vicuna) and we tested several configurations focusing on the *findings generation* task.

In this document, we describe details of several configurations tested, including different vision encoders (CLIP and BiomedCLIP), VL projector (matrix and MLP) and language model for text decoding (LLaMA1.0 and Vicuna-7b). Among our findings, we highlight that integrating the output of another method as input context for LLaVA resulted in our best version for the challenge.

2 Task Description

2.1 Datasets

The data provided by the challenge (Xu et al., 2024) consists of 5 datasets PadChest (Bustos et al., 2020), BIMCV-COVID19 (Vayá et al., 2020), CheXpert (Chambon et al., 2024), OpenI, and MIMIC-CXR (Johnson et al., 2019). All of them have a medical report with at least the finding section, in total, we have $344,394$ training samples.

In the present work, we focus only on the finding generation, in each training step we use the findings section, of the official train datasets. We do not use any extra dataset or data augmentation techniques.

The results reported in this work are measured in the challenge hidden test set which has $1,063$ samples for the generation of the finding section.

3 Methodology

3.1 Model Architecture

The architecture used in this work is known as Large Language and Vision Assistant for

[1] https://huggingface.co/dcampanini

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 608–613
August 16, 2024. ©2024 Association for Computational Linguistics

BioMedicine (LLaVA-Med Li et al., 2023), we fine-tuned this system following different approaches described in the section 3.2.

The LLaVA-Med system has 3 main blocks (Figure 1), the first is a vision encoder, then a vision-language connector to project the image features into the word embedding space, and finally, a Large Language Model (LLM) that processes visual and language tokens to generate a final answer.

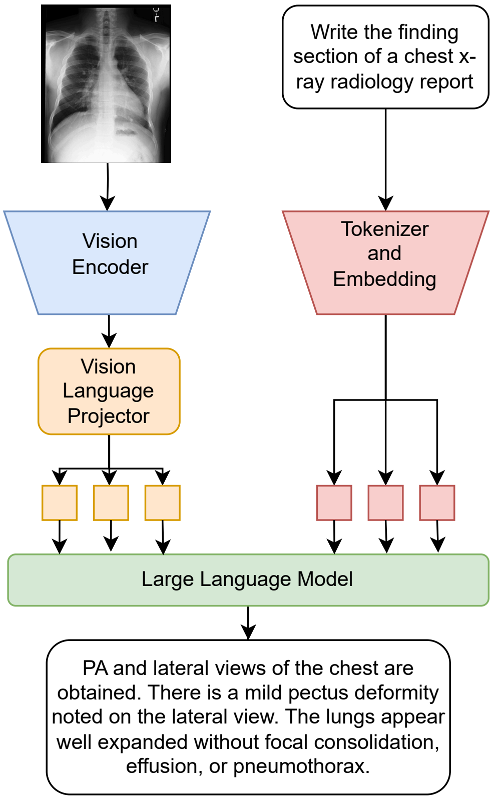

Write the finding section of a chest x-ray radiology report
Vision Encoder
Tokenizer and Embedding
Vision Language Projector
Large Language Model
PA and lateral views of the chest are obtained. There is a mild pectus deformity noted on the lateral view. The lungs appear well expanded without focal consolidation, effusion, or pneumothorax.

Figure 1: LLaVA-Med architecture used in this work. The Large Language Model (LLM) processes the features extracted from the image and the prompt.

There are different options for each block, in our case, we use 3 different vision encoders all of them based on CLIP (Radford et al., 2021), they are clip-vit-L-patch14, clip-vit-large-patch14-336, and BiomedCLIP (Zhang et al., 2023). For the connector, we choose a projection matrix and a 2 layer MLP. Finally, for the LLM we select LLaMA1.0-7B, and Vicuna-7b-v1.5. Table 1 summarizes the 3 model versions used during the challenge.

3.2 Training Strategy

We train our models in 2 stages, similar to the strategy proposed in Liu et al., 2023, but adapted for report generation, and not for instruction tuning. The stages are detailed as follows:

- **Stage 1 or alignment:** the image encoder and the LLM are frozen, and the MLP or projection matrix are trained.
- **Stage 2 or fine-tuning:** the MLP or projection matrix and the LLM are trained.

For both stages, we train with samples formed by one image and the respective finding section. Our models process one image at a time. Therefore, we manipulate the training dataset when more than one image is associated with a medical report.

For the dataset MIMIC-CXR, for each medical report we select the Anterior Posterior (AP) image or the Posterior Anterior (PA) image, and the finding section.

We use the image's name to select the frontal images for CheXpert, which indicates the view presented in the X-ray exam (frontal or lateral).

For the last 3 datasets PadChest, BIMCV-COVID19, and OpenI, we take the first image in the array of images associated with each medical report, which was, in general, a frontal view.

For the Model-1.0 (Table 1) we start the fine-tuning from a LLaVA-Med checkpoint shared in the official GitHub repository[2], and we update the linear matrix and the LLM using different combinations of the official train datasets. Stage 1 is omitted for this model since the based model was trained in biomedical data extracted from PMC-15M (Zhang et al., 2023) an image-text dataset extracted from scientific publications.

For Model-1.1 and Model-1.2 (Table 1) we train following the 2 stages strategy, for the stage 1 we use the complete challenge train dataset, considering only one image per finding section, we employ more training samples in this stage since is more general than stage 2, so more broad data can help the final model performance.

On the other hand, for stage 2, we only use MIMIC-CXR. This decision is discussed in the section 4. For these 2 models, we have to execute stage 1, since we don't have a projector specialized in medicine to connect the vision-encoder with the LLM embedding space.

For stage 1 we always use a learning rate of 1×10^{-3} and a cosine learning rate with a warmup ratio of 3%. Similarly, for stage 2 we employ the same scheduling and warmup ratio but with a learning rate of 1×10^{-4}. Every stage is performed in a GPU NVIDIA RTX A6000 with 48 GB of memory.

[2] https://github.com/microsoft/LLaVA-Med

Model version	Vision-encoder	VL Projector	LLM
Model-1.0	clip-vit-L-patch14	Matrix	LLaMA1.0-7b
Model-1.1	clip-vit-large-patch14-336	MLP	Vicuna-7b-v1.5
Model-1.2	BiomedCLIP	MLP	Vicuna-7b-v1.5

Table 1: Configuration of the 3 model architectures used during the challenge.

3.3 Text Prompt

The prompt given to the LLMs performs an important role in getting a solid model performance. In our work, the LLM can receive as input the image feature and the prompt.

For stage 1, we follow the strategy mentioned in Chaves et al., 2024 and we use only the image to train the projector, no prompt or extra information is provided to the model, in this way, we force the LLM to focus on the images.

On the other hand, for stage 2, the prompt is formed by the model context and the instruction, with the first we can control for example the model personality, asking to be polite, and in our case, we also define that the LLM does not have to provide dates, hours or text with enumeration in the report. The final instruction to the LLM is: *Write the finding section of a chest x-ray radiology report*. The complete prompt (context + instruction) is described in the following paragraph:

- **Context:** *You are LLaVA-Med, a large language and vision assistant. Write in the style of a radiologist, write one fluent text without enumeration, dates, or hours of the day, be concise, and don't provide explanations.*

- **Instruction:** *Write the finding section of a chest x-ray radiology report.*

The previously described prompt is used in stage 2 and inference.

Additionally, we have considered making another test, improving the prompt using as extra information other findings sections, generated by a multilabel classifier and a group of templates (Pino et al., 2021). This different system consists of a DenseNet-121 CNN trained to classify 13 pathologies for chest X-ray images, and then using the output labels, we generate the finding section based on a group of template sentences. The LLM receives as input the image features and the new prompt with extra information, which we call enriched-prompt. The new prompt instruction looks as follows:

- **Instruction:** *Write the finding section of a chest x-ray radiology report using the image, and the following information: the lungs are clear. heart size is normal the cardiomediastinal silhouette is normal. there is noted left sided or right sided , small, moderate, or large pneumothorax in the lung no pleural effusions. there is no evidence of fibrosis no displaced fracture is seen there is a noted right sided or left sided picc or tube*

4 Experiments and Results

In Table 2 we report the results of the 3 model versions trained with different dataset configurations and performing or not stage 1. All results are only for the finding generation task.

The metrics outline in Table 2 are BLEU4 (B4 Papineni et al., 2002), ROUGEL (RL Lin, 2004), Bertscore (BS Zhang et al., 2019), F1-cheXbert (F1-cXb Smit et al., 2020), and F1-RadGraph (F1-RG Delbrouck et al., 2022a), the last column represents the average between these metrics. All the values are calculated using the official leaderboard web page with the framework VilMedic (Delbrouck et al., 2022b).

The first result in Table 2 is for the Model-1.0 without any posterior finituning or training, which is the original LLaVa-Med shared in the official repository, it has a poor performance generating finding. It is by far our worst model, so it should be fine-tuned to get good results even in tasks inside the biomedical domain.

From our experiments with Model-1.0, we see that considering only MIMIC-CXR we have good enough results comparable to using MIMIC-CXR + CheXpert, and consistently outperforming the same Model-1.0 trained with the complete challenge datasets (Table 2). For this reason, the training of the other models is performed only employing MIMIC-CXR for stage 2.

When we make use of BiomedClip (Zhang et al., 2023) we see a clear improvement in 6.31 percentual points for F1-cheXbert in comparison with the second-best model in this metrics (29.37 vs

Stage1	Ep	Stage2	Ep	B4	RL	BS	F1-cXb	F1-RG	Avg
Model-1.0 Clip LLaMA1.0-7B									
None	0	None	0	0.95	11.69	27.79	15.46	4.15	12.01
None	0	MIMIC-CXR	1	4.67	19.58	48.74	18.00	16.29	21.46
None	0	MIMIC-CXR	3	5.05	19.13	47.51	23.06	15.77	**22.10**
None	0	MIMIC-CXR + CheXpert	1	4.78	19.19	47.57	20.25	15.47	21.45
None	0	All	1	3.24	15.45	42.12	18.11	11.79	18.14
Model-1.1 Clip-336 Vicuna1.5-7B									
All	1	MIMIC-CXR	1	5.16	19.68	47.92	11.61	16.78	20.23
All	1	MIMIC-CXR	3	3.92	19.75	48.06	5.92	16.06	18.74
Model-1.1 Clip-336 Vicuna1.5-7B Enriched-prompt									
All	1	MIMIC-CXR	1	**6.46**	**20.51**	**49.23**	9.35	**18.59**	20.83
Model-1.2 BiomedClip Vicuna1.5-7B									
All	1	MIMIC-CXR	1	3.48	16.31	35.49	**29.37**	15.51	20.03

Table 2: Results on the hidden test set, for all 3 model versions without applying enriched-prompt, and the Model-1.1 improved through the enriched-prompt technique. All numbers are calculated using Vilmedic on the official challenge web page.

Model	B4	RL	BS	F1-cXb	F1-RG	Avg
Model-1.1 Clip-336 Vicuna1.5-7B	5.16	19.68	47.92	11.61	16.78	20.23
DenseNet-121 classifier + templates	4.81	15.96	44.03	**33.69**	18.41	23.38
Model-1.1 Clip-336 Vicuna1.5-7B + enriched prompt from DenseNet-121 classifier	**6.46**	**20.51**	**49.23**	9.35	**18.59**	20.83

Table 3: Efects of the enrich-prompt technique. The last row represents the metrics of the resulting system, which is the Model-1.1 but enhanced with the enriched prompt coming from the DenseNet-121+templates system.

Model	B4	RL	BS	F1-cXb	F1-RG	Avg
Model-1.1 Clip-336 Vicuna1.5-7B	5.16	19.68	47.92	11.61	16.78	20.23
DenseNet-121 classifier + templates v1	4.81	15.96	44.03	**33.69**	18.41	**23.38**
Model-1.1 Clip-336 Vicuna1.5-7B + DenseNet-v1	**6.46**	20.51	49.23	9.35	18.59	20.83
DenseNet-121 classifier + templates v2	4.74	16.17	47.28	27.44	13.08	21.74
Model-1.1 Clip-336 Vicuna1.5-7B + DenseNet-v2	5.94	19.40	47.20	7.15	16.87	19.31
DenseNet-121 classifier + templates v3	5.50	17.11	48.97	26.26	14.47	22.46
Model-1.1 Clip-336 Vicuna1.5-7B + DenseNet-v3	5.10	19.98	48.94	8.11	17.47	19.92
DenseNet-121 classifier + templates v4	4.18	17.05	42.91	27.20	**19.42**	22.15
Model-1.1 Clip-336 Vicuna1.5-7B + DenseNet-v4	5.21	**20.80**	**50.14**	5.90	18.51	20.11

Table 4: Impact of the enriched prompt technique using different template models, and the same multimodal model highlighted in gray. The resulting model's metrics are pointed out in yellow.

23.06). This suggests that the feature extracted from the image with this vision encoder allows to the model classify properly more pathologies than the previous vision encoders, considering that F1-cheXbert is a metric focus in the classification of 14 labels.

We apply the enriched-prompt technique to the model with the best F1-RadGraph, which is the *Model-1.1 Clip-336 Vicuna1.5-7B*, the result of employing this procedure is an improvement in BLEU-4, Rouge-L, Bert-Score, and F1-RadGraph, but a big fall in F1-cheXbert (Table 2, 3), this indicates that the model is not good at classifying the 14 classes considered by the metric.

Table 3 shows the change in the metrics for 2 base models, combined across the prompt. When we apply in-context learning to the Model-1.1 Clip-336 Vicuna1.5-7B adding to the prompt the reports generated by the DenseNet-121+templates, the resulting model overcomes the metrics of both previous systems, except for F1-cheXpert.

Another consequence of implementing an enriched prompt is the generation of shorter findings in comparison with those generated by the other model versions and by the classifier plus template sentences.

In Table 4 we show more evidence of the enriched prompt technique. The most consistent effect over the two base models is observed in the F1-RadGraph metric, improving up to 1.81 percentage points (pp) in the base multimodal model, and up to 3.79 pp for the template models. For Rouge-L, and Bert-Score we can also see an enhancement in the based models, the most outstanding result is the increase of 7.23 pp in Bert-Score for the DenseNet-121 classifier + templates v4. The different versions of the template models consider distinct types of templates and classifier hyperparameters, more detail about it can be found in the paper of iHealth-Chile-3&2. On the other hand, the effect of the enriched prompt technique in F1-cheXbert is always a big fall.

5 Conclusion

In this work, we performed an analysis of different model architectures based on LLaVA-Med, we conclude that using the best possible vision-encoder, and LLM we can improve some specific aspects of the system, such as the NLP overlapping (BLEU-4 and Rouge-L) or the more classification related metrics (F1-cheXbert), nevertheless to see more consistent results we suggest that more quality data is needed, particularly for alignment (stage 1). Moreover, since the promising results in F1-cheXbert obtained with BiomedCLIP is convenient to develop a vision-encoder custom to x-ray images.

Finally, the enriched-prompt techniques show auspicious results. It can work as a guide for the LLM, it shows good metrics when we calculate BLEU-4, Rouge-L, BertScore, and F1-RadGraph, but it should be complemented with an accurate classifier system to improve the F1-cheXbert.

Limitations

There are some limitations in the system that we propose. For instance, our model is unable to use multiple images, however, the medical reports for chest x-rays are usually formed by two or three views of the patient chest, so we are missing potentially important information.

The quality of the medical report generated with the enriched prompt technique should be analyzed in more depth, especially because of the large drop in the F1-cheXbert metric.

Another limitation is that our approach is computationally expensive, which limits the quantity of experiments that we can perform. Finally, our reports are not hallucinations free, for example in some cases, the model generates findings referring to another report for the same patient, but this is a problem because the model does not know previous patient exams.

Acknowledgements

This work has been funded by Millennium Institute for Intelligent Healthcare Engineering (iHEALTH) ICN2021_004; National Center for Artificial Intelligence (CENIA) FB210017, Basal ANID; Fondecyt 1221425; and the National Agency for Research and Development (ANID) through the Scholarship Program / Doctorado Becas Chile / 2019 - 21191569.

References

Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria De La Iglesia-Vaya. 2020. Padchest: A large chest x-ray image dataset with multi-label annotated reports. *Medical image analysis*, 66:101797.

Pierre Chambon, Jean-Benoit Delbrouck, Thomas Sounack, Shih-Cheng Huang, Zhihong Chen, Maya Varma, Steven QH Truong, Chu The Chuong, and Curtis P. Langlotz. 2024. Chexpert plus: Augmenting a large chest x-ray dataset with text radiology reports, patient demographics and additional image formats.

Juan Manuel Zambrano Chaves, Shih-Cheng Huang, Yanbo Xu, Hanwen Xu, Naoto Usuyama, Sheng Zhang, Fei Wang, Yujia Xie, Mahmoud Khademi, Ziyi Yang, Hany Awadalla, Julia Gong, Houdong Hu, Jianwei Yang, Chunyuan Li, Jianfeng Gao, Yu Gu, Cliff Wong, Mu Wei, Tristan Naumann, Muhao Chen, Matthew P. Lungren, Serena Yeung-Levy, Curtis P. Langlotz, Sheng Wang, and Hoifung Poon. 2024. Towards a clinically accessible radiology foundation model: open-access and lightweight, with automated evaluation.

Jean-Benoit Delbrouck, Pierre Chambon, Christian Bluethgen, Emily Tsai, Omar Almusa, and Curtis Langlotz. 2022a. Improving the factual correctness of radiology report generation with semantic rewards. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4348–4360.

Jean-benoit Delbrouck, Khaled Saab, Maya Varma, Sabri Eyuboglu, Pierre Chambon, Jared Dunnmon, Juan Zambrano, Akshay Chaudhari, and Curtis Langlotz. 2022b. Vilmedic: a framework for research at the intersection of vision and language in medical ai. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 23–34.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. 2023. LLaVA-med: Training a large language-and-vision assistant for biomedicine in one day. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *NeurIPS*.

Pablo Messina, Pablo Pino, Denis Parra, Alvaro Soto, Cecilia Besa, Sergio Uribe, Marcelo Andía, Cristian Tejos, Claudia Prieto, and Daniel Capurro. 2022. A survey on deep learning and explainability for automatic report generation from medical images. *ACM Computing Surveys (CSUR)*, 54(10s):1–40.

Maram Mahmoud A Monshi, Josiah Poon, and Vera Chung. 2020. Deep learning in generating radiology reports: A survey. *Artificial Intelligence in Medicine*, 106:101878.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Pablo Pino, Denis Parra, Cecilia Besa, and Claudio Lagos. 2021. Clinically correct report generation from chest x-rays using templates. In *Machine Learning in Medical Imaging*, pages 654–663, Cham. Springer International Publishing.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision.

Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Y Ng, and Matthew Lungren. 2020. Combining automatic labelers and expert annotations for accurate radiology report labeling using bert. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1500–1519.

Eric Topol. 2019. *Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again*, 1st edition. Basic Books, Inc., USA.

Maria De La Iglesia Vayá, Jose Manuel Saborit, Joaquim Angel Montell, Antonio Pertusa, Aurelia Bustos, Miguel Cazorla, Joaquin Galant, Xavier Barber, Domingo Orozco-Beltrán, Francisco García-García, et al. 2020. Bimcv covid-19+: a large annotated dataset of rx and ct images from covid-19 patients. *arXiv preprint arXiv:2006.01174*.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Sheng Zhang, Yanbo Xu, Naoto Usuyama, Jaspreet Bagga, Robert Tinn, Sam Preston, Rajesh Rao, Mu Wei, Naveen Valluri, Cliff Wong, Matthew Lungren, Tristan Naumann, and Hoifung Poon. 2023. Large-scale domain-specific pretraining for biomedical vision-language processing.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

iHealth-Chile-3&2 at RRG24: Template Based Report Generation

Oscar Loch [1,2,3], Pablo Messina [1,2,3], Rafael Elberg [1,2,3],
Diego Campanini [2], Álvaro Soto [1,3], René Vidal [4], and Denis Parra [1,2,3]
[1] Department of Computer Science, Pontifical Catholic University of Chile.
[2] Millennium Institute for Intelligent Healthcare Engineering (iHEALTH), Chile.
[3] National Center for Artificial Intelligence (CENIA), Chile.
[4] University of Pennsylvania.
{oscar.loch,pamessina,rafael.elberg}@uc.cl, diego.campanini@ing.uchile.cl,
vidalr@seas.upenn.edu, {asoto,dparra}@ing.puc.cl

Abstract

This paper presents the approaches of the iHealth-Chile-3 and iHealth-Chile-2 teams for the shared task of Large-Scale Radiology Report Generation at the BioNLP workshop. Inspired by prior work on template-based report generation, both teams focused on exploring various template-based strategies, using predictions from multi-label image classifiers as input. Our best approach achieved a modest F1-RadGraph score of 19.42 on the findings hidden test set, ranking 7th on the leaderboard. Notably, we consistently observed a discrepancy between our classification metrics and the F1-CheXbert metric reported on the leaderboard, which always showed lower scores. This suggests that the F1-CheXbert metric may be missing some of the labels mentioned by the templates.

1 Introduction

The generation of radiology reports (RRG) from medical imaging using deep learning represents a significant area of ongoing research (Messina et al., 2022). Successfully implementing this task can help reduce the workload and time spent on administrative duties, such as composing text reports. This efficiency enables physicians to focus more on patient interaction (Topol, 2019) and in identifying anomalies from multiple input images.

There is a pressing need for eXplainable AI (XAI) (Gunning et al., 2019) in critical domains like medicine. In the context of report generation, the explainability aspect remains understudied (Messina et al., 2022). Some models address this issue by generating saliency maps that highlight important pixels, using techniques such as Grad-CAM (Selvaraju et al., 2019) for CNN networks or visualizing attention maps for Transformer networks. However, some authors argue against relying solely on saliency maps as explanations. For instance, Rudin (2019) advocates for using inherently interpretable models that are constrained by domain knowledge, making them transparent and understandable for humans.

To enhance transparency and understandability of our implementation in the Shared task (Xu et al., 2024), we use a simple template-based report generation model. Specifically, we reimplement and modify the template-based strategy proposed by Pino et al. (2021). The team iHealth-Chile-3 focused on meticulously reproducing Pino et al.'s approach, employing DenseNet-121 and a conventional multilabel classification layer for 13 CheXpert classes (excluding "No Findings"), as shown in Figure 2. Meanwhile, team iHealth-Chile-2 developed a different image classifier that combines DenseNet-121 with text embeddings of factual statements, which can be both classified and visually grounded, leveraging very recent work on fact extraction and encoding from radiology reports (Messina et al., 2024). This approach, shown in Figure 3, can be seen as a more general version of stage 1 of CheXfusion (Kim, 2023), the winning method in the ICCV CVAMD 2023 Shared Task on CXR-LT: Multi-Label Long-Tailed Classification on Chest X-Rays (Holste et al., 2023).

2 Task Description

2.1 Datasets

The data provided by the challenge consists of five datasets: PadChest (Bustos et al., 2020), BIMCV-COVID19 (Vayá et al., 2020), CheXpert (Chambon et al., 2024), OpenI (Demner-Fushman et al., 2016), and MIMIC-CXR (Johnson et al., 2019). Each of these datasets includes radiology reports paired with at least one image. The entire training set comprises $344,394$ reports with at least the Findings section and $366,413$ reports with at least the Impression section. Additionally, the challenge permitted the use of VinDr-CXR (Nguyen et al., 2022), which contains $18,000$ frontal chest X-ray images with labels and bounding box annotations,

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 614–623
August 16, 2024. ©2024 Association for Computational Linguistics

but no reports.

In this participation, iHealth-Chile-3 focused on training using only MIMIC-CXR and CheXpert, utilizing the CheXpert labels (Irvin et al., 2019) from both datasets. For training the CNN, this team only used the 13 labels associated with findings (excluding the "No Findings" label) and treated the uncertain label (-1) as negative (0). iHealth-Chile-3 did not employ any additional datasets or data augmentation techniques.

On the other hand, iHealth-Chile-2 leveraged concurrent work on fact extraction and encoding from radiology reports, which includes 591,920 factual statements extracted from MIMIC-CXR radiology reports. A representative subset of these facts was sampled, and with the assistance of a Natural Language Inference (NLI) system (the explanation of which is beyond the scope of this paper), negative facts were identified for all the reports. Furthermore, by combining 78 classes from the Chest ImaGenome dataset (Wu et al., 2021) and the 26 classes from the CXR-LT 2023 challenge (Holste et al., 2023) and removing the overlap, a total of 93 classes were exhaustively annotated by the same NLI system, providing more standardized supervision for MIMIC-CXR. iHealth-Chile-2 also utilized CheXpert, with the 14 classes adapted as short factual statements, VinDr-CXR, with its 28 classes adapted for fact classification, and the 22 bounding box classes used for visual grounding supervision. OpenI was also adapted for fact classification by converting its manual and automatic tags into short sentences with the assistance of GPT-4.

The results reported in this work are measured using the challenge's hidden test set, which contains $1,063$ samples for the generation of the Findings section.

3 Methodology

3.1 Model Architecture

The approaches followed by both teams are summarized in Figure 1. Essentially, an image classifier is trained for multi-label classification. This classifier is then used to make predictions over one or more views, which are processed by a rule-based algorithm to build the final report. Both teams used the PyTorch implementation of DenseNet-121 (Huang et al., 2017) as the visual backbone of their models, outputting 1024-D feature vectors.

The specific implementation by iHealth-Chile-3 is shown in Figure 2. This approach strictly follows Pino et al.'s straightforward implementation (Pino et al., 2021). A fully connected layer predicts 13 classes. For each classified label, there is a pair of fixed sentences: one for when the label is classified as present and another for when it is absent. These sentences are then concatenated to form the final report.

In contrast, iHealth-Chile-2 replaces the fully connected layer with a more sophisticated attention-based pooling mechanism conditioned on a fact embedding, as shown in Figure 3. This approach has the added advantage that the attention can be supervised with ground-truth visual grounding annotations if available, such as bounding boxes in the case of VinDr-CXR. Furthermore, its use of text embeddings to indicate the fact to classify allows the model to work as an *open-vocabulary* multi-label classifier, which can be easily applied to an arbitrary number of datasets with different number of classes or factual statements.

3.2 Training Strategy and Implementation Details

iHealth-Chile-3. This team trained models on MIMIC-CXR and CheXpert using CheXpert labels, selecting the first image in the array of images associated with each medical report, which was generally a frontal view.

To address class imbalance, a Weighted Binary Cross Entropy Loss was employed. The model was optimized using Adam with a learning rate of 0.0001 and a weight decay of 0.00001. Additionally, a learning rate scheduler reduced the learning rate by a factor of 0.1 if the monitored metric did not improve for three consecutive epochs. This dynamic adjustment helps refine the training process and achieve better convergence based on the model's performance. The input images were resized to 256×256 and normalized with a mean and standard deviation of 0.5.

The model was trained for 12 epochs with a batch size of 110, using an NVIDIA RTX A6000 GPU, with an estimated training time of 42 hours.

iHealth-Chile-2. This team utilized the MIMIC-CXR, CheXpert, VinDr-CXR, and OpenI datasets. To ensure a more balanced sampling of all datasets in subsequent batches, a multi-dataset dataloader was implemented. This dataloader sampled from each dataset with a weight of 5.0 for MIMIC-CXR and 1.0 for each of the other datasets, giving

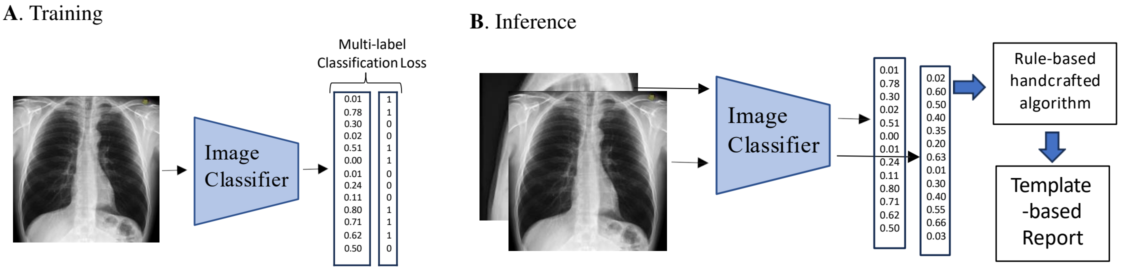


Figure 1: Overview of the template-based approach followed by both teams. During training, a single-view image classifier is trained for multi-label classification. During inference, the image classifier is used to predict labels for one or all the views associated with a given report to generate. These classification predictions are then processed by a handcrafted rule-based algorithm that builds the final report.

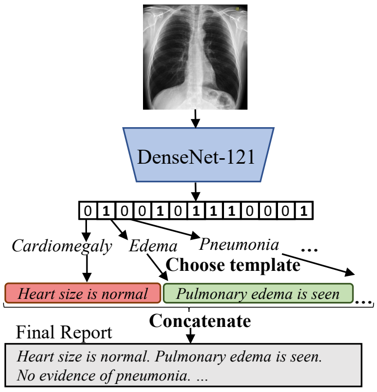


Figure 2: Template-Based Architecture of iHealth-Chile-3. The DenseNet is trained to classify the 13 labels as shown in the scheme. After the training is complete, in inference, the DenseNet is frozen and classifies the 13 labels for an input image. For each label, a template sentence is chosen depending on the absence or presence of the label. Finally, the chosen template sentences are concatenated into a final report.

MIMIC-CXR more weight due to its larger number of facts to classify, as discussed in Section 2.1.

For CheXpert and VinDr-CXR, a hybrid loss combining standard BCE, Weighted by Class BCE, and Focal Loss was used because these datasets have a fixed number of classes. For MIMIC-CXR and OpenI, BCE + Focal Loss was employed. In the case of VinDr-CXR, the Mean Absolute Error (MAE) between the predicted attention map and the ground-truth bounding boxes is used as attention supervision loss for visual grounding of the classified facts. The AdamW optimizer (Loshchilov and Hutter, 2019) was used with a cyclic exponential learning rate varying from 1e-4 to 1e-6 over 8 epochs. Each epoch consisted of approximately 800 batches. The model was trained for about 20 hours, after which no significant gains in validation metrics were observed. The batch size was 13 images per batch, with about 40 facts sampled per image. Combined with 10 gradient accumulation steps, the effective batch size was 130 images. Images were resized to 416 × 416.

All experiments were implemented using Python 3.10.10 with PyTorch version 1.13.1+cu117 (Paszke et al., 2017). The experiments were conducted on a computing node equipped with a 20-core Intel(R) Core(TM) i9-9900X CPU @ 3.50GHz, three NVIDIA GPUs—two GeForce RTX 2080 Ti with 11GB memory and one GeForce RTX 3090 with 24GB memory. The system was complemented by 125GB of RAM.

3.3 Report Generation using Templates

For report generation, curated sets of two sentences per abnormality were manually selected to indicate presence and absence. These sets are categorized into different types of templates (Pino et al., 2021): *Mimic Style*, *Ambiguous*, *Fusion*, and *Fusion + Groups*.

The *Mimic Style* sentences correspond to a simple template shown in Appendix Table 5, while the *Ambiguous* sentences correspond to the template shown in Appendix Table 6. On the other hand, the *Fusion* template combines the absent template sentences from *Mimic Style* with the present template sentences from *Ambiguous*.

The *Fusion + Groups* template functions differently from the other templates. Instead of replacing a sentence for each label, it groups labels together. If a group of labels matches the value of abnormalities specified in a grouped template (see Appendix Table 7), that template is added to the final report. After iterating through all grouped templates, the remaining abnormalities are addressed using the *Fusion* template for each individual disease, thus giving the template its name *Fusion + Groups*.

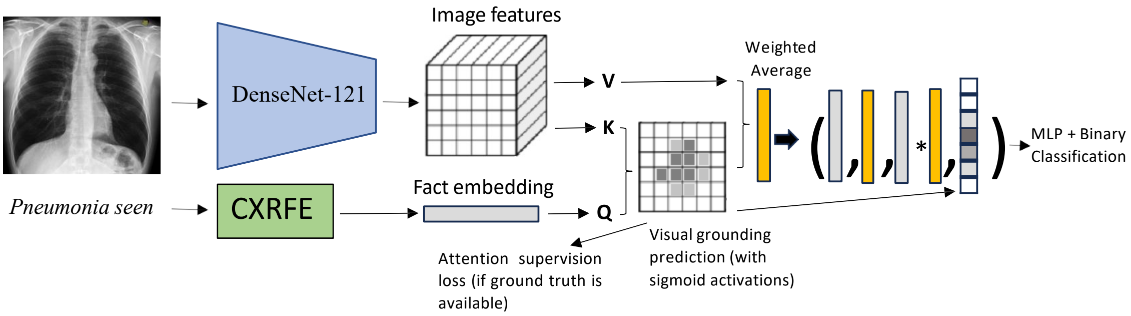


Image features
DenseNet-121
V
K
Q
Weighted Average
MLP + Binary Classification
Pneumonia seen
CXRFE
Fact embedding
Attention supervision loss (if ground truth is available)
Visual grounding prediction (with sigmoid activations)

Figure 3: Fact Classifier architecture of iHealth-Chile-2. CXRFE stands for **C**hest **X**-**R**ay **F**act **E**ncoder, an improved version of CXR-BERT (Boecking et al., 2022) via several NLP tasks, as outlined in a concurrent publication (Messina et al., 2024). This Fact Classifier is an experimental architecture, that seeks to generalize the stage-1 classifier of CheXfusion (Kim, 2023). Unlike iHealth-Chile-3, the Fact Classifier is trained on all views, and during inference the predictions from all views are aggregated via max-pooling.

Table 1: Classification metrics on the MIMIC-CXR and CheXpert validation sets using the CNN trained by iHealth-Chile-3.

Precision	Recall	F1-Micro	F1-Macro
0.36	0.74	0.48	0.36

4 Experiments and Results

iHealth-Chile-3. After training the CNN, we obtained the classification results shown in Table 1. We achieved a precision of 0.36, which, being relatively low, immediately impacts our performance on the NLP metrics discussed later in this section. Furthermore, the significantly lower value of F1-Macro compared to F1-Micro suggests that the model performs notably weaker on specific labels, likely due to class imbalance.

Table 2 presents the results of report generation on the findings and impression hidden test sets. The metrics detailed are BLEU4 (B4 Papineni et al., 2002), ROUGE-L (RL Lin, 2004), BERTScore (BS Zhang et al., 2019), F1-CheXbert (chX Smit et al., 2020), and F1-RadGraph (RG Delbrouck et al., 2022a). All values were calculated using the official leaderboard web page with the VilMedic framework (Delbrouck et al., 2022b). By examining Table 2, we can observe that the Template Type which most increases the F1-RadGraph score is the *Ambiguous* Template type, improving this score by at least 2 points compared to the *Mimic Style* Template. This improvement is likely due to the inclusion of location-specific terms like "left" and "right." However, there is a corresponding decrease in BLEU4, possibly because the ground-truth report specifies the location of the disease, and the addition of terms like "left" and "right" might introduce inaccuracies.

Additionally, Table 2 reveals that the best Template for the findings section, based on F1-RadGraph, is the *Fusion* + *Groups* template, while for the impression section, the best is the *Fusion* Template.

On the other hand, the F1-CheXbert score is lower than the F1-Macro and F1-Micro scores for the classification of CheXpert labels. This suggests that the BERT model used for the F1-CheXbert metric may not accurately detect some of the labels encoded in the template-generated sentences, even if they are simple, making this metric potentially unreliable for this task. A similar issue is observed with BERTScore, which does not consistently align with the other metrics.

iHealth-Chile-2. Table 3 presents the classification and template-based report generation metrics on the MIMIC-CXR and CheXpert validation sets. We highlight two notable results from this Table: (1) The Fact Classifier achieves significantly higher scores when evaluated with labels produced by the same tool used to annotate the training set (i.e., VisualCheXbert for CheXpert and the NLI labeler for MIMIC-CXR); and (2) The performance drops when the CheXpert labeler and CheXbert evaluate a template-based report built from the classifications, particularly with F1-CheXbert (macro and micro). This provides further evidence that the metric may be missing some of the labels mentioned in the templates.

Additional evidence of the impact of the labeling tool on the evaluation is provided in Appendix Table 8. One evaluation considers 78 classes from the

Table 2: iHealth-Chile-3's metrics on the hidden test sets. All metrics are calculated using Vilmedic on the official challenge web page. The abbreviations used are: B4 (BLEU4), RL (ROUGE-L), BS (Bertscore), cXb (F1-cheXbert), and RG (F1-RadGraph).

Template Type	Findings Hidden Test Set					Impression Hidden Test Set				
	B4	RL	BS	cXb	RG	B4	RL	BS	cXb	RG
Mimic Style	4.74	16.17	**47.28**	27.44	13.08	**1.72**	9.41	36.18	24.55	8.30
Ambiguous	3.58	14.65	44.99	**29.35**	15.85	1.64	9.84	**37.38**	**26.84**	10.34
Fusion	**4.80**	16.88	46.73	28.20	18.70	1.66	**10.21**	37.21	25.82	**11.58**
Fusion + Groups	4.18	**17.05**	42.91	27.20	**19.42**	1.42	10.13	33.01	24.91	11.53

Table 3: Classification and Template-based Report Generation results on the validation sets of MIMIC-CXR and CheXpert. The classes considered are the 14 classes of the CheXpert dataset. On MIMIC-CXR we consider two sources of ground-truth labels for evaluation: the CheXpert labeler and our own NLI labeler. In the case of CheXpert, we use the labels produced by VisualCheXbert (Jain et al., 2021) that were released with the dataset. The reports were produced with the *Fusion + Groups* technique.

Classification: CheXpert labeler / VisualCheXbert				**Classification: NLI labeler (ours)**				**Template-based Report Generation**			
F1 (micro)	**F1 (macro)**	**PRC-AUC (micro)**	**PRC-AUC (macro)**	**F1 (micro)**	**F1 (macro)**	**PRC-AUC (micro)**	**PRC-AUC (macro)**	**F1-CheXp (micro)**	**F1-CheXp (macro)**	**F1-CheXb (micro)**	**F1-CheXb (macro)**
MIMIC-CXR validation set (9178 images)											
0.491	0.405	0.418	0.416	0.628	0.519	0.668	0.557	0.510	0.424	0.430	0.372
CheXpert validation set (5468 images)											
0.679	0.554	0.719	0.717	-	-	-	-	0.539	0.417	0.442	0.358

Table 4: iHealth-Chile-2's metrics on the findings-hidden-test-set and impression-hidden-test-set.

Dataset	**Method**	**B4**	**RL**	**BS**	**cXb**	**RG**
findings-hidden-test-set	Fact Classifier + Templates (*Fusion + Groups*)	**4.81**	**15.96**	**44.03**	**33.69**	**18.41**
findings-hidden-test-set	Fact Classifier + BART (findings, v1)	2.33	14.22	43.39	28.00	14.48
findings-hidden-test-set	Fact Classifier + BART (findings, v2)	2.78	14.29	43.40	31.00	14.74
impression-hidden-test-set	Fact Classifier + BART (impression)	2.28	11.33	35.98	20.87	7.59

Chest ImaGenome dataset (Wu et al., 2021), while the other considers the 26 classes from the CXR-LT 2023 challenge (Holste et al., 2023). Noticeably, the performance drops significantly when evaluated with the original labels compared to the labels generated by our NLI system. This discrepancy suggests that either our NLI system is incorrect, or the labels provided by the original datasets, which were also extracted from reports, are inaccurate. This issue warrants further investigation in future work.

Lastly, Table 4 presents all submissions by iHealth-Chile-2 to the hidden test set (findings and impression). The best approach is clearly based on templates. However, for completeness, we also include unsuccessful attempts at producing reports generatively using BART (Lewis et al., 2020), a sequence-to-sequence model, by training it to generate reports from templates. This approach degraded performance, so we advise against it.

5 Conclusions and Future Work

We have presented the results of the iHealth-Chile-3 and iHealth-Chile-2 teams in the Large-Scale Radiology Report Generation shared task. Both teams used a template-based method, where an image classifier predicts specific classes, which are then used to generate a report with predefined templates. The performance in the challenge was modest. Interestingly, despite the templates being tailored for CheXpert classes, the F1-CheXbert metrics were consistently lower than the classification metrics.

Based on these results, future work should focus on: (1) Thoroughly evaluating report generation metrics to identify and address limitations in existing ones; (2) Improving chest X-ray image classifiers, particularly for long-tail classes; and (3) Developing more advanced report generation systems that surpass rigid templates while preserving classifier accuracy for long-tail classes.

6 Limitations

The iHealth-Chile-3's approach has several limitations that warrant discussion. Firstly, this approach is restricted in its ability to specify the location of detected abnormalities. It can only confirm the presence or absence of these abnormalities without providing detailed localization within the images. This spatial limitation may affect clinical applicability, where precise localization is often critical.

Secondly, the overall performance of the reports generated by this approach is inherently tied to the performance of the multi-label classifier employed. Any deficiencies or inaccuracies in the classifier directly impact the quality and reliability of the generated reports. Moreover, even if the multi-label classifier were to achieve perfect performance, the scope of the reports would still be confined to the 13 specific labels used in this approach. This means that any abnormalities outside these predefined categories would go unreported, potentially missing other clinically significant findings.

Additionally, the resolution of the images used in this study, limited to 256x256 pixels, could further constrain the performance. Lower resolution images may lack the necessary detail for accurate detection and classification of certain abnormalities, leading to potential misclassification or oversight. Future work could explore the impact of using higher resolution images to determine if this enhances the diagnostic accuracy and overall utility of the approach.

The strategy adopted by iHealth-Chile-2 has notable limitations as well. Firstly, it is based on an experimental architecture still under development and unpublished at the time of this writing. It also depends on an auxiliary Natural Language Inference (NLI) system that is being developed concurrently, with significant involvement of GPT-4. As discussed in Section 4, the discrepancies between the original labels from source datasets and our NLI-based labels highlight the need for further investigation. We aim to elaborate on these aspects in future publications.

The Fact Classifier tested by iHealth-Chile-2 may also be limited by its use of DenseNet-121 as its visual backbone. Given the advances in architectures based on vision transformers, such as the Swin Transformer (Liu et al., 2021), DenseNet-121 might not be the optimal choice. This limitation is also shared by iHealth-Chile-3.

Lastly, a significant limitation in the classification approach itself followed by both teams is the lack of a clear strategy for translating classifications into a final natural language report. Even if an optimal open-vocabulary classifier were to accurately identify a comprehensive list of abnormalities with good visual grounding, it remains unclear how to convert these predictions into a report that scores well according to the challenge metrics. This gap between classification/visual grounding and report generation warrants further investigation.

Acknowledgements

This work has been funded by Millenium Science Initiative Program ICN2021_004 (iHEALTH); IMFD ICN17 002, and Fondecyt grants 11230762 and 1231724; National Center for Artificial Intelligence (CENIA) FB210017, Basal ANID; Fondecyt 1221425; and the National Agency for Research and Development (ANID) through the Scholarship Program / Doctorado Becas Chile / 2019 - 21191569.

References

Benedikt Boecking, Naoto Usuyama, Shruthi Bannur, Daniel C Castro, Anton Schwaighofer, Stephanie Hyland, Maria Wetscherek, Tristan Naumann, Aditya Nori, Javier Alvarez-Valle, et al. 2022. Making the most of text semantics to improve biomedical vision–language processing. In *European conference on computer vision*, pages 1–21. Springer.

Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria De La Iglesia-Vaya. 2020. Padchest: A large chest x-ray image dataset with multi-label annotated reports. *Medical image analysis*, 66:101797.

Pierre Chambon, Jean-Benoit Delbrouck, Thomas Sounack, Shih-Cheng Huang, Zhihong Chen, Maya Varma, Steven QH Truong, Chu The Chuong, and Curtis P. Langlotz. 2024. Chexpert plus: Augmenting a large chest x-ray dataset with text radiology reports, patient demographics and additional image formats.

Jean-Benoit Delbrouck, Pierre Chambon, Christian Bluethgen, Emily Tsai, Omar Almusa, and Curtis Langlotz. 2022a. Improving the factual correctness of radiology report generation with semantic rewards. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4348–4360.

Jean-benoit Delbrouck, Khaled Saab, Maya Varma, Sabri Eyuboglu, Pierre Chambon, Jared Dunnmon, Juan Zambrano, Akshay Chaudhari, and Curtis Langlotz. 2022b. Vilmedic: a framework for research at the intersection of vision and language in medical ai. In *Proceedings of the 60th Annual Meeting of the*

Association for Computational Linguistics: System Demonstrations, pages 23–34.

Dina Demner-Fushman, Marc D Kohli, Marc B Rosenman, Sonya E Shooshan, Laritza Rodriguez, Sameer Antani, George R Thoma, and Clement J McDonald. 2016. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association*, 23(2):304–310.

David Gunning, Mark Stefik, Jaesik Choi, Timothy Miller, Simone Stumpf, and Guang-Zhong Yang. 2019. Xai—explainable artificial intelligence. *Science Robotics*, 4(37):eaay7120.

Gregory Holste, Song Wang, Ajay Jaiswal, Yuzhe Yang, Mingquan Lin, Yifan Peng, and Atlas Wang. 2023. Cxr-lt: Multi-label long-tailed classification on chest x-rays. *PhysioNet*, 5:19.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. 2017. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, Jayne Seekins, David A. Mong, Safwan S. Halabi, Jesse K. Sandberg, Ricky Jones, David B. Larson, Curtis P. Langlotz, Bhavik N. Patel, Matthew P. Lungren, and Andrew Y. Ng. 2019. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison.

Saahil Jain, Akshay Smit, Steven QH Truong, Chanh DT Nguyen, Minh-Thanh Huynh, Mudit Jain, Victoria A Young, Andrew Y Ng, Matthew P Lungren, and Pranav Rajpurkar. 2021. Visualchexbert: addressing the discrepancy between radiology report labels and image labels. In *Proceedings of the Conference on Health, Inference, and Learning*, pages 105–115.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317.

Dongkyun Kim. 2023. Chexfusion: Effective fusion of multi-view features using transformers for long-tailed chest x-ray classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2702–2710.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Pablo Messina, Pablo Pino, Denis Parra, Alvaro Soto, Cecilia Besa, Sergio Uribe, Marcelo Andía, Cristian Tejos, Claudia Prieto, and Daniel Capurro. 2022. A survey on deep learning and explainability for automatic report generation from medical images. *ACM Computing Surveys (CSUR)*, 54(10s):1–40.

Pablo Messina, René Vidal, Denis Parra, Álvaro Soto, and Vladimir Araujo. 2024. Extracting and encoding: Leveraging large language models and medical knowledge to enhance radiological text representation.

Ha Q Nguyen, Khanh Lam, Linh T Le, Hieu H Pham, Dat Q Tran, Dung B Nguyen, Dung D Le, Chi M Pham, Hang TT Tong, Diep H Dinh, et al. 2022. Vindr-cxr: An open dataset of chest x-rays with radiologist's annotations. *Scientific Data*, 9(1):429.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.

Pablo Pino, Denis Parra, Cecilia Besa, and Claudio Lagos. 2021. Clinically correct report generation from chest x-rays using templates. In *Machine Learning in Medical Imaging*, pages 654–663, Cham. Springer International Publishing.

Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.*, 1(5):206–215.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2019. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359.

Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Y Ng, and Matthew Lungren. 2020. Combining automatic labelers and expert annotations for accurate radiology report labeling using bert. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1500–1519.

Eric Topol. 2019. *Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again*, 1st edition. Basic Books, Inc., USA.

Maria De La Iglesia Vayá, Jose Manuel Saborit, Joaquim Angel Montell, Antonio Pertusa, Aurelia Bustos, Miguel Cazorla, Joaquin Galant, Xavier Barber, Domingo Orozco-Beltrán, Francisco García-García, et al. 2020. Bimcv covid-19+: a large annotated dataset of rx and ct images from covid-19 patients. *arXiv preprint arXiv:2006.01174*.

Joy T Wu, Nkechinyere Agu, Ismini Lourentzou, Ismini Lourentzou, Arjun Sharma, Joseph Alexander Paguio, Jasper Seth Yao, Edward C Dee, William Mitchell, Satyananda Kashyap, Andrea Giovannini, Leo Anthony Celi, and Mehdi Moradi. 2021. Chest imagenome dataset for clinical reasoning. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1. Curran.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and “discharge me!”. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Appendix

A.1 Templates used by Health-Chile-3’s approach

The *Mimic Style* template set, shown in Table 5, corresponds to sentences which simply indicate presence or absence of the labels. This template set was named *Mimic Style* because the sentences were chosen manually to imitate the sentences found in the MIMIC-CXR reports.

The *Ambiguous* template set, shown in Table 6, corresponds to sentences which when they indicate presence are ambiguous. For example, they can be ambiguous in terms of location, indicating the presence of an abnormality on the left or right side of the image.

Finally, the *Group* template set (not to be confused with the *Fusion + Groups* template approach) serves as an auxiliary template to be combined with the simpler templates that indicate the single presence of labels. This template set is shown in detail in Table 7.

Table 5: Sentences in the Mimic Style template set.

Abnormality	Absence template	Presence template
Cardiomegaly	Heart size is normal	The heart is enlarged
Enlarged Cardiomed.	The mediastinal contour is normal	The cardiomediastinal silhouette is enlarged
Consolidation	No focal consolidation	There is focal consolidation
Lung Opacity	The lungs are free of focal airspace disease	One or more airspace opacities are seen
Atelectasis	No atelectasis	Appearance suggest atelectasis
Pleural Effusion	No pleural effusion	Pleural effusion is seen
Pleural Other	No fibrosis	Pleural thickening is present
Pneumonia	No pneumonia	There is evidence of pneumonia
Pneumothorax	No pneumothorax is seen	There is pneumothorax
Edema	No pulmonary edema	Pulmonary edema is seen
Lung Lesion	No pulmonary nodules or mass lesions identified	There are pulmonary nodules or mass identified
Fracture	No fracture is seen	A fracture is identified
Support Devices	-	A device is seen

Table 6: Sentences in the Ambiguous template set.

Abnormality	Absence template	Presence template
Cardiomegaly	no cardiomegaly	the heart is stable, mild, moderate, severe or enlarged in size
Enlarged Cardiomed.	mediastinal contour is normal	the cardiomediastinal silhouette is unchanged, enlarged or widened
Consolidation	no consolidation	there is observed left or right lung consolidation
Lung Opacity	free of focal airspace disease	there are left or right present lung airspace opacities
Atelectasis	no atelectasis	there is observed left or right lung present atelectasis
Pleural Effusion	no pleural effusion	there is an observed left, right or bilateral, small, moderate or large pleural effusion
Pleural Other	no fibrosis	there is present left or right, minimal, mild or severe pleural thickening
Pneumonia	no pneumonia	observed process left or right lung pneumonia
Pneumothorax	no pneumothorax	there is noted left sided or right sided, small, moderate or large pneumothorax in the lung
Edema	no pulmonary edema	there is noted mild, moderate or severe pulmonary edema
Lung Lesion	no pulmonary nodules	there are left or right pulmonary lung nodules observed
Fracture	no fracture	there is a rib or clavicular left or right sided fracture
Support Devices	there is no picc line	there is a noted right sided or left sided picc or tube

Table 7: Sentences for Group Template.

Abnormalities	Value of labels	Template Group Sentence
’Lung Lesion’, ’Lung Opacity’, ’Edema’, ’Consolidation’, ’Pneumonia’, ’Atelectasis’	0 (all absent)	the lungs are clear
’Consolidation’, ’Pleural Effusion’, ’Pneumothorax’	0 (all absent)	there is no focal consolidation , pleural effusion , or pneumothorax .
’Pneumothorax’, ’Pleural Effusion’	0 (all absent)	there is no pleural effusion or pneumothorax .
’Pneumothorax’, ’Consolidation’	0 (all absent)	there is no focal consolidation or pneumothorax .

Table 8: Fact classification results on MIMIC-CXR test set. These results are shown for illustrative purposes only. The performance achieved by the fact classifier according to the labels produced by our NLI labeler is significantly higher than the performance according to the original labeling tools of the datasets.

Original Labeler		**NLI labeler**	
F1 (micro)	**F1 (macro)**	**F1 (micro)**	**F1 (macro)**
CXR-LT (26 classes)			
0.451	0.306	0.620	0.454
Chest ImaGenome (78 classes)			
0.321	0.261	0.533	0.355

Gla-AI4BioMed at RRG24: Visual Instruction-tuned Adaptation for Radiology Report Generation

Xi Zhang, Zaiqiao Meng[*]**, Jake Lever** and **Edmond S. L. Ho**
School of Computing Science, University of Glasgow
X.Zhang.6@research.gla.ac.uk, Zaiqiao.Meng@glasgow.ac.uk
Jake.Lever@glasgow.ac.uk, Shu-Lim.Ho@glasgow.ac.uk

Abstract

We introduce a radiology-focused visual language model designed to generate radiology reports from chest X-rays. Building on previous findings that large language models (LLMs) can acquire multimodal capabilities when aligned with pretrained vision encoders, we demonstrate similar potential with chest X-ray images. This integration enhances the ability of model to understand and describe chest X-ray images. Our model combines an image encoder with a fine-tuned LLM based on the Vicuna-7B architecture, enabling it to generate different sections of a radiology report with notable accuracy. The training process involves a two-stage approach: (i) initial alignment of chest X-ray features with the LLM (ii) followed by fine-tuning for radiology report generation[1].

1 Introduction

Radiology reports constitute the primary medium through which radiologists convey the findings and conclusions derived from radiography, such as chest X-rays. These reports play a pivotal role in the diagnostic and therapeutic processes across a wide range of diseases, emphasizing their significance in contemporary medical practice (Engle et al., 2021). Structured to enhance clarity and efficacy in medical communication, radiology reports primarily feature FINDINGS and IMPRESSIONS sections (Kahn et al., 2009). The FINDINGS section details the critical observations of the radiologist on the image, while the IMPRESSIONS section summarizes the conclusions and recommendations of the radiologist. These sections collectively ensure that radiology reports are indispensable in diagnostic and therapeutic decision-making, combining image analysis and clinical insight. Table 1 shows an example generated by GPT-4 (OpenAI et al., 2024), which delineates these sections.

FINDINGS
There has been an increase in size of the left pleural effusion compared to the prior exam. The right lung remains clear with no evidence of consolidation or pneumothorax. The heart size is mildly enlarged but stable. The mediastinum appears unremarkable. Mild degenerative changes are noted in the thoracic spine and ribs. The upper abdomen is without remarkable findings.
IMPRESSIONS
Increase in left pleural effusion compared to prior. Stable mild cardiomegaly. No evidence of right lung pathology.

Table 1: FINDINGS and IMPRESSIONS in a synthetic radiology report generated by GPT-4.

Radiology report generation (RRG) is crucial for advancing future medical artificial intelligence systems (Monshi et al., 2020). This task involves transforming images into text, necessitating alignment between imaging and textual data. Significant advancements in natural language processing have driven progress in this area, with large generative visual language models like LLaVA (Liu et al., 2023), InstructBLIP (Dai et al., 2023), and Flamingo (Alayrac et al., 2022) leading the way.

The prevailing visual language models, such as those mentioned above, aim to address the challenge of multimodal alignment by leveraging large-scale pretraining. Typically, this involves adapting a vision encoder for integration with a pretrained LLM. To meet specific task requirements, various degrees of finetuning are applied. For example, LLaVA (Liu et al., 2023) represents a novel end-to-end trained large multimodal model for general-purpose visual and language understanding, achieving impressive chat capabilities. However, in the context of our work, the focus is on fine-tuning

[*] Corresponding author.

[1] https://github.com/Glasgow-AI4BioMed/RRG-BioNLP-ACL2024.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 624–634
August 16, 2024. ©2024 Association for Computational Linguistics

image-text pairs, specifically for tasks related to medical images, to enhance the capability of the visual language model in radiology report generation.

In the specialized area of radiographic report generation, it is paramount for models to discern nuanced details within multiple medical images. These details include subtle variations in opacity against a backdrop of overlapping structures (Panayides et al., 2020). Therefore, the radiographic report generation task extends beyond the mere extraction of details from a single image. Models must interpret the clinical implications of these nuances to generate precise and medically rigorous text in reports. This is a particularly crucial capability for radiographic report generation models, since it enhances the clinical utility and effectiveness of radiology reports and ensures their accuracy and relevance in clinical settings.

General-domain models have proven inadequate for generating findings in radiology reports (Hyland et al., 2024). In this work, we propose a radiology-specific visual language model designed for solving the radiology report generation task by fine-tuning across various sections of medical reports. Our model utilizes a two-stage fine-tuning process that significantly enhances its performance. In particular, we initially align the large language model with image embedding through a pretraining phase. In the second stage, we further fine-tune the LLM using Low-Rank adaptation (LoRA) techniques (Hu et al., 2021). Both stages are trained on the dataset in this workshop (Xu et al., 2024).

Additionally, we use a straightforward strategy of merging and stitching multiple images to form a single cohesive input, enabling the model to effectively process and integrate information from multiple X-ray images. Using the dataset provided by this workshop, which includes a collection of chest X-rays and their corresponding sections, we fine-tune our model to enhance the accuracy and specificity of the generated radiology reports.

This paper investigates the fine-tuning of visual instruction for a visual language model in the specific domain of radiology report generation. We describe the training of two distinct models developed for the Shared Task on Large-Scale Radiology Report Generation (RRG24) at the BioNLP 2024 Workshop (Xu et al., 2024). In the public test set, we achieved an F1-RadGraph score (Delbrouck et al., 2022a) of 24.13 and 22.79 in the Findings and Impressions sections, respectively. In the hidden test set, we achieved F1-RadGraph scores (Delbrouck et al., 2022a) of 24.13 and 22.10 in the Findings and Impressions sections, respectively, which places us 4th on the leaderboard at the time of submission. The contributions of this research are as follows:

- We enhance domain adaptation for radiology by implementing visual instruction tuning, which further fine-tunes the visual language model specifically for image-to-text tasks. This approach optimizes performance in interpreting and translating visual data into descriptive, clinically relevant text.

- We adopt a method of stitching multiple images together, allowing a single image encoder to process multiple image inputs simultaneously. This strategy obviates the need for separate encoding of each image, enabling the model to adapt to varying numbers of image inputs using limited resources.

2 Related Work

Nowadays, exemplified by open-source projects such as LLaVA (Liu et al., 2023), the effectiveness of self-supervised vision-language models (VLMs) using parallel data has been demonstrated in different research domains. These VLMs, when instruction-tuned with multimodal inputs, align well with human intentions and perform robustly in various downstream tasks, including converting images to text (Park and Kim, 2023).

However, the unique characteristics of biomedical image-text pairs significantly differ from those in general domains. Biomedical images often contain subtle and complex features that require precise interpretation, while the corresponding text must convey highly specific medical information (Huff et al., 2021). In biomedical settings, VLMs designed for general domains often fail to meet these specialized needs, as they lack the ability to accurately interpret medical data and generate relevant clinical descriptions (Chang et al., 2023). This discrepancy underscores the urgent need for domain-specific fine-tuning. By tailoring VLMs to the distinct demands of the biomedical field, such fine-tuning can enhance their ability to capture and convey the intricate details necessary for accurate medical interpretations and reports.

Recent advancements have been made in adapting general-purpose foundation models for medical applications, particularly in radiology. The

Med-Flamingo (Moor et al., 2023), an extension of the OpenFlamingo framework (Awadalla et al., 2023), leverages images and captions from medical textbooks to enhance few-shot visual question-answering capabilities. Similarly, the Med-PaLM M, developed by Tu et al. (2023), fine-tuned the PaLM-E model (Driess et al., 2023) using comprehensive biomedical datasets. LLaVA-Med, proposed by Li et al. (2023), modifies the LLaVA (Liu et al., 2023) framework with image-text pairings and multimodal instructions from PubMed data. Additionally, the ELIXR model, developed by Xu et al. (2023), integrates the SupCon CXR encoder (Sellergren et al., 2022) with the PaLM 2-S model (Anil et al., 2023) to support classification, semantic search, question answering, and quality assurance. Finally, the Radiology-GPT, created by Liu et al. (2024), utilizes radiology reports from MIMIC-CXR (Johnson et al., 2019) to facilitate the generation of findings-to-impression text, based on the Alpaca instruction-tuning framework (Taori et al., 2023).

Historically, research in radiology report generation has varied, with some studies focusing exclusively on either the Findings or the Impressions sections (Jin et al., 2024; Yan et al., 2023), while others have addressed both. Notably, Endo et al. (2021) and Bannur et al. (2023) specialized in generating only the Impressions section. In contrast, studies by Miura et al. (2021), Delbrouck et al. (2022a), Tanida et al. (2023), Nicolson et al. (2023), and Tu et al. (2023) concentrated on the Findings section. Comprehensive analyses by Yu et al. (2023) and Jeong et al. (2023) covered all settings, demonstrating that the choice of sections significantly influences reported performance metrics, complicating comparative evaluations across different study designs.

However, these existing models have limitations. Most notably, they are typically designed to process single images and often fall short in generating reports that match the depth and detail of those written by human radiologists. Additionally, they do not fully replicate the workflow of medical professionals, who often reference multiple images to enhance report accuracy. Our work addresses these gaps by developing a model capable of handling multiple images simultaneously and generating comprehensive radiology reports. This approach aims to more closely mimic the process used by medical professionals, thereby improving the accuracy and quality of the generated reports.

3 Methodology

In our study, we follow the observations from LLaVA-Med (Li et al., 2023), suggesting superior performance when initiating with a language-only pretrained LLM rather than a multimodal-trained base. Our model architecture incorporates an image encoder and a learnable adapter placed atop the image outputs, mirroring the LLaVA-1.5 model design (Liu et al., 2023). We adopt an auto-regressive language modelling approach using cross-entropy loss (Graves, 2014) and align hyperparameters with those from LLaVA-1.5, including a joint tuning phase for the LLM and adapter (Liu et al., 2023). In alignment with LLaVA-1.5 protocols, we initially pretrain the adapter alone for one epoch, followed by a full training cycle lasting three epochs, employing Low-Rank Adaptation of Large Language Models techniques (LoRA) (Hu et al., 2021) for efficient parameter tuning.

3.1 Task Description

A key application of natural language generation in medicine is developing support systems that produce written reports from X-ray images, detailing clinical findings. Such systems are highly valuable, potentially reducing the routine workload of radiologists and improving the efficacy of clinical interactions. The objective of this shared task is to generate radiology reports from one or more chest X-rays taken during a single study, specifically targeting two sections: 'Findings' and 'Impressions' (as shown in Table 1).

Consequently, our team is dedicated to the task of exclusively producing either the 'Findings' or 'Impressions' sections of the report. We have developed separate models for each section because their focuses are different. The 'Findings' section provides a factual description based on the images, while the 'Impressions' section offers the radiologist's conclusions and recommendations. By separating the models, we can tailor each to better address its specific requirements.

For the radiology report section generation, handling multiple images is crucial as it allows the model to provide a detailed and accurate description of the observed facts, similar to how radiologists analyze multiple images to form a comprehensive understanding. This approach enhances the model's ability of to mimic the actual workflow of medical professionals, who often reference multiple images.

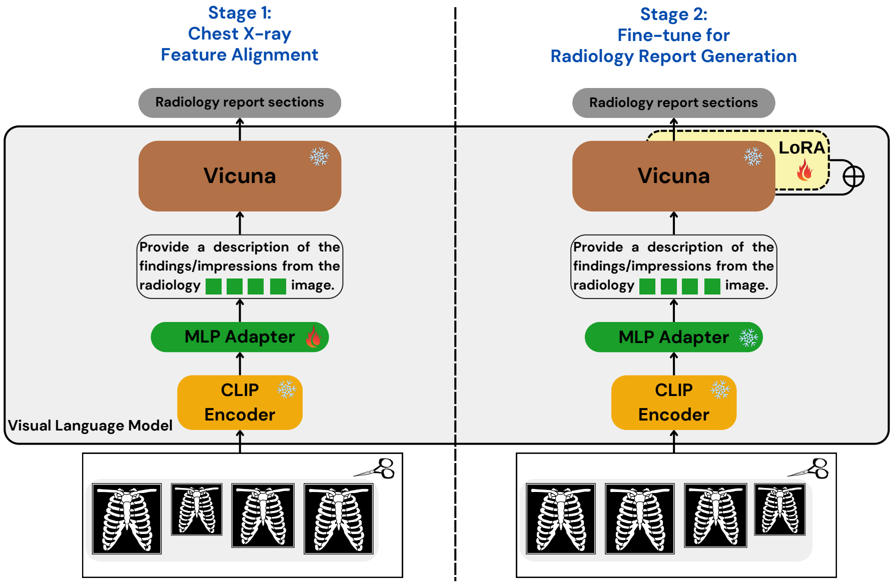

Stage 1:
Chest X-ray
Feature Alignment
Radiology report sections
Vicuna
Provide a description of the findings/impressions from the radiology 🟩🟩🟩🟩 image.
MLP Adapter
CLIP Encoder
Visual Language Model
Stage 2:
Fine-tune for
Radiology Report Generation
Radiology report sections
LoRA
Vicuna
Provide a description of the findings/impressions from the radiology 🟩🟩🟩🟩 image.
MLP Adapter
CLIP Encoder

Figure 1: Our two-stage training framework. In the first stage, visual features are aligned with LLM. In the second stage, the model focuses on the training of radiology report generation tasks.

3.2 The Proposed Models

To solve the RRG24 shared task, we fine-tune a large visual language model for radiology report generation based on the provided dataset. Specifically, we propose two separate models, called Med-CXRGen-F and Med-CXRGen-I, fine-tuned on Findings and Impressions sections respectively.

We use CLIP (Radford et al., 2021) as an image encoder and Vicuna-1.5 (Chiang et al., 2023) as a large language model. Our adaptation module consists of a multi-layer perceptron (MLP) featuring GELU activations (Hendrycks and Gimpel, 2023) and a uniform hidden size of 1024 across all layers.

The interaction with the model involves alternating system messages linked with the corresponding image. The training objective of the model is to generate accurate responses. Initially, we convert the image into a series of image patch tokens via the image encoder, selecting embeddings from the penultimate layer. These image features are then processed by the MLP adapter, aligning them to the input specifications of the LLM.

The instructional prompt of the report generation task we employed is: "Provide a description of the findings/impressions from the radiology <image>\n image." In this prompt, "<image>\n" represents the image holder token, as shown in Figure 1, which indicates to the LLM that it should base its generation on the input image.

3.3 Training

The same network architecture is utilized for different radiology report sections, where an MLP adapter connects the vision encoder and the language model. For model training, we use a two-stage procedure: (as shown in Figure 1)

- **Stage 1: Chest X-ray Feature Alignment** In the first epoch training phase on the provided dataset, each sample, accompanied by instructions and image input, prompts the model to predict the original caption. During this stage, we keep the visual encoder and LLM weights unchanged, focusing solely on updating the MLP adapter. This approach aligns the features from chest X-ray images with their textual embeddings in the LLM. Training is limited to a single epoch, which facilitates the expansion of the vocabulary of aligned image-text tokens specific to the radiology domain.

- **Stage 2: Fine-tune for Radiology Report Generation**
  In the second phase, the visual encoder weights and adapter are kept frozen while continuing to update the pre-trained LLM weights using LoRA (Hu et al., 2021) technology. Further fine-tuning is conducted on the provided dataset through visual instrumental tuning with three epochs.

4 Evaluation

4.1 Dataset

We fine-tune and evaluate our models using the RRG24 dataset hosted on the BioNLP ACL'24 (Xu et al., 2024), which includes data from MIMIC-CXR (Johnson et al., 2019), CheXpert (Chambon et al., 2024), PadChest (Bustos et al., 2020), BIMCV-COVID19 (de la Iglesia Vayá et al., 2020), and OpenI, with their statistics shown in Table 2.

Dataset	FINDINGS	IMPRESSIONS
training	344,394	366,413
validation	8,839	9,331
test-public	2,692	2,967
test-hidden	1,063	1,428

Table 2: Distribution of shared task on Large-Scale Radiology Report Generation.

We conducted training in two stages (refer to section 3.3). To ensure consistency between training and inference processes, we analysed the word count distribution, as shown in Table 3. Consequently, we have set a maximum length of 1024 for both the text input and inference output to minimise computational expense. On the other hand, as illustrated in Table 4, some datasets contain multiple images, therefore, we select up to the first four images for the image input. We merge multiple images horizontally to form a single-image input, which is proven to be robust in our experiments.

Dataset	FINDINGS	IMPRESSIONS
training	259 (±180)	216 (±153)
validation	257 (±176)	217 (±155)
test-public	380 (±161)	257 (±224)

Table 3: Average word count and standard deviation on Large-Scale Radiology Report Generation.

Dataset	FINDINGS	IMPRESSIONS
training	1.57 (±0.63)	1.45 (±0.62)
validation	1.58 (±0.62)	1.45 (±0.62)
test-public	1.70 (±0.71)	1.67 (±0.71)

Table 4: Average number of images and standard deviation on Large-Scale Radiology Report Generation.

4.2 Metrics

We assess the generated reports through a dual approach involving both general lexical metrics and specialized radiology metrics. Focusing on the accuracy of described medical findings, radiology-specific metrics provide a deeper insight into the clinical relevance of the reports, beyond surface-level phrasing variations. According to the RRG24 guidelines, we consider five evaluation metrics for this work, including BLEU4 (Papineni et al., 2002), ROUGEL (Lin, 2004), BERT score (Zhang et al., 2020), F1-cheXbert (Smit et al., 2020), and F1-RadGraph (Delbrouck et al., 2022a).

4.3 Training details

We evaluated our two proposed models, i.e. Med-CXRGen-F and Med-CXRGen-I, on the workshop evaluation datasets, based on a computational infrastructure utilizing an A6000 GPU (48GB memory each) with the Deepspeed zero-3 configuration (Rajbhandari et al., 2020) with BF16 enabled. We employ a cosine learning rate scheduler that begins with a warm-up phase of 0.03 and sets the learning rate at $1 \cdot 10^{-5}$. The global batch size for our experiments is set at 16. Observations of the smallest loss on the evaluation dataset throughout the training process guide us to select this as the final checkpoint for all runs. For inference on the test dataset, we decode in 32-bit precision up to 150 tokens, consistent with the baseline model on the leaderboard (Delbrouck et al., 2022b). Each model required approximately 215 hours of training.

5 Results

We report the performance of our two proposed models over five evaluation metrics in Table 5. As shown in Table 5, in the public test set, we achieved an F1-RadGraph score (Delbrouck et al., 2022a) of 24.13 and 22.79 in the Findings and Impressions sections, respectively. In the hidden test set, we achieved F1-RadGraph scores (Delbrouck et al., 2022a) of 24.13 and 22.10 in the Findings and

Model	Dataset	Section	BLEU4	ROUGEL	Bertscore	F1-cheXbert	F1-RadGraph
Med-CXRGen-F	validation	Findings	7.02	23.33	48.93	40.42	21.94
	test-public	Findings	8.07	24.90	53.45	45.91	24.13
	test-hidden	Findings	7.65	24.35	52.69	46.21	24.13
Med-CXRGen-I	validation	Impressions	10.18	28.10	51.78	50.51	26.65
	test-public	Impressions	7.10	25.11	47.39	47.43	22.79
	test-hidden	Impressions	9.60	25.27	48.60	46.74	22.10

Table 5: Evaluation results on different datasets.

Impressions sections, respectively, which places us 4th on the leaderboard[2] at the time of submission. Additionally, our model achieved notable Bertscore results in the test-public set, with 53.45 for Findings and 47.39 for Impressions. These results demonstrate the effectiveness of our approach in generating high-quality medical reports across different datasets.

6 Discussion

Performance disparities observed between the Findings and Impressions sections of the test results can be attributed to several factors. Firstly, the Impression and Findings sections address distinct medical purposes, resulting in performance disparities. The Findings section offers an objective description of symptoms, while the Impressions are oriented towards diagnostic conclusions. The variability in word count between these sections also affects the complexity of model inference, as reflected in the lexical evaluation scores.

Additionally, significant discrepancies in the medical evaluation metrics highlight a varied distribution of diseases within the test set. This heterogeneity could impact the generalisability and accuracy of the model. Furthermore, our analysis indicates that the performance may be compromised in multi-image inference scenarios where it does not account for superfluous images. Such factors are essential to consider when assessing the diagnostic accuracy and reliability of the model in clinical settings. Enhancing the ability of model to differentiate between relevant and superfluous images could significantly improve diagnostic accuracy.

Furthermore, exploring domain-specific adaptations and fine-tuning strategies tailored to the unique characteristics of medical data could further enhance model performance. Incorporating temporal dynamics into the model to capture changes over time and developing more sophisticated frameworks for generating multi-modal radiology reports are other promising avenues for future research. These advancements are expected to enhance both the practicality and accuracy of our model within clinical scenarios.

7 Conclusion

In this work, we have developed a vision-language model capable of processing multiple images. Through visual instruction tuning, we achieved alignment between two modalities and further fine-tuning for specific downstream tasks. Notably, our system attained a commendable fourth-place standing across four diverse test datasets at the RRG24 at BioNLP 2024 workshop (Xu et al., 2024), substantiating the practicality of vision-language models within specialized medical tasks.

Moving forward, we intend to conduct in-depth research into more sophisticated methods for generating multi-modal radiology reports. This will involve incorporating temporal dynamics and developing frameworks specifically focused on text generation. Such advancements are expected to enhance both the practicality and accuracy of our model within the clinical scenario.

[2] https://vilmedic.app/misc/bionlp24/leaderboard

8 Limitations

The following section outlines the limitations identified in our study:

1. **Prevalence of certain conditions:** Some diseases are more easily detected, which may lead to artificially high medical assessment scores.

2. **Imaging modalities and anatomical structures:** There is a notable imbalance in the imaging modalities and anatomical structures covered in the training dataset. Variations such as the number of images per patient and the considerable disparity in the length of medical reports exacerbate this imbalance.

3. **Radiologist and radiology department preferences:** Preferences and writing styles vary among radiologists and radiology departments. This diversity adds complexity to medical reports by introducing inconsistencies and uncertainties that are, to a certain extent, human-induced. For example, the dataset provided in this workshop demonstrates that even the same radiology section descriptions have varying styles. These elements significantly complicate the task of report generation.

These limitations highlight areas for improvement and the need for methodological refinements to enhance model effectiveness and reliability in clinical environments. These challenges were not addressed within the scope of this workshop.

Ethics Statement

The concepts and methodologies presented in this paper are based on experimental research findings. They are not currently available for other use. The data used for the training processes were exclusively sourced from the provided dataset, which complies with ethical standards regarding Patient Health Information. Adherence to these standards guarantees the responsible handling of sensitive data throughout our research.

Acknowledgments

We extend our gratitude to the BioNLP 2024 RRG24 Shared Task organizers for providing the baseline pipeline ViLMedic (Delbrouck et al., 2022b) and curating these challenging and exciting tasks (Xu et al., 2024).

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. 2022. Flamingo: a visual language model for few-shot learning. *Preprint*, arXiv:2204.14198.

Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou, Denny Zhou, Slav Petrov, and Yonghui Wu. 2023. Palm 2 technical report. *Preprint*, arXiv:2305.10403.

Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. 2023. Openflamingo: An open-source framework for training large autoregressive vision-language models. *Preprint*, arXiv:2308.01390.

Shruthi Bannur, Stephanie Hyland, Qianchu Liu, Fernando Pérez-García, Maximilian Ilse, Daniel C. Castro, Benedikt Boecking, Harshita Sharma, Kenza Bouzid, Anja Thieme, Anton Schwaighofer, Maria Wetscherek, Matthew P. Lungren, Aditya Nori, Javier Alvarez-Valle, and Ozan Oktay. 2023. Learning to exploit temporal structure for biomedical vision-language processing. *Preprint*, arXiv:2301.04558.

Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria de la Iglesia-Vayá. 2020. Padchest: A large chest x-ray image dataset with multi-label annotated reports. *Medical Image Analysis*, 66:101797.

Pierre Chambon, Jean-Benoit Delbrouck, Thomas Sounack, Shih-Cheng Huang, Zhihong Chen, Maya Varma, Steven QH Truong, Chu The Chuong, and Curtis P. Langlotz. 2024. Chexpert plus: Augmenting a large chest x-ray dataset with text radiology reports, patient demographics and additional image formats. *Preprint*, arXiv:2405.19538.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. 2023. A survey on evaluation of large language models. *Preprint*, arXiv:2307.03109.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. 2023. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Maria de la Iglesia Vayá, Jose Manuel Saborit, Joaquim Angel Montell, Antonio Pertusa, Aurelia Bustos, Miguel Cazorla, Joaquin Galant, Xavier Barber, Domingo Orozco-Beltrán, Francisco García-García, Marisa Caparrós, Germán González, and Jose María Salinas. 2020. Bimcv covid-19+: a large annotated dataset of rx and ct images from covid-19 patients. *Preprint*, arXiv:2006.01174.

Jean-Benoit Delbrouck, Pierre Chambon, Christian Bluethgen, Emily Tsai, Omar Almusa, and Curtis Langlotz. 2022a. Improving the factual correctness of radiology report generation with semantic rewards. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4348–4360, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jean-benoit Delbrouck, Khaled Saab, Maya Varma, Sabri Eyuboglu, Pierre Chambon, Jared Dunnmon, Juan Zambrano, Akshay Chaudhari, and Curtis Langlotz. 2022b. ViLMedic: a framework for research at the intersection of vision and language in medical AI. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 23–34, Dublin, Ireland. Association for Computational Linguistics.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. 2023. Palm-e: An embodied multimodal language model. *Preprint*, arXiv:2303.03378.

Daisuke Endo, Ryota Kobayashi, Ramon Bartolo, Bruno B Averbeck, Yasuko Sugase-Miyamoto, Kazuko Hayashi, Kenji Kawano, Barry J Richmond, and Shigeru Shinomoto. 2021. A convolutional neural network for estimating synaptic connectivity from spike trains. *Scientific Reports*, 11(1):12087.

Ryann L Engle, David C Mohr, Sally K Holmes, Marjorie Nealon Seibert, Melissa Afable, Jenniffer Leyson, and Mark Meterko. 2021. Evidence-based practice and patient-centered care: doing both well. *Health care management review*, 46(3):174–184.

Alex Graves. 2014. Generating sequences with recurrent neural networks. *Preprint*, arXiv:1308.0850.

Dan Hendrycks and Kevin Gimpel. 2023. Gaussian error linear units (gelus). *Preprint*, arXiv:1606.08415.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Daniel T Huff, Amy J Weisman, and Robert Jeraj. 2021. Interpretation and visualization techniques for deep learning models in medical imaging. *Physics in Medicine & Biology*, 66(4):04TR01.

Stephanie L. Hyland, Shruthi Bannur, Kenza Bouzid, Daniel C. Castro, Mercy Ranjit, Anton Schwaighofer, Fernando Pérez-García, Valentina Salvatelli, Shaury Srivastav, Anja Thieme, Noel Codella, Matthew P. Lungren, Maria Teodora Wetscherek, Ozan Oktay, and Javier Alvarez-Valle. 2024. Maira-1: A specialised large multimodal model for radiology report generation. *Preprint*, arXiv:2311.13668.

Jaehwan Jeong, Katherine Tian, Andrew Li, Sina Hartung, Fardad Behzadi, Juan Calle, David Osayande, Michael Pohlen, Subathra Adithan, and Pranav Rajpurkar. 2023. Multimodal image-text matching improves retrieval-based chest x-ray report generation. *Preprint*, arXiv:2303.17579.

Haibo Jin, Haoxuan Che, Yi Lin, and Hao Chen. 2024. Promptmrg: Diagnosis-driven prompts for medical report generation. *Preprint*, arXiv:2308.12604.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317.

Charles E. Kahn, Curtis P. Langlotz, Elizabeth S. Burnside, John A. Carrino, David S. Channin, David M. Hovsepian, and Daniel L. Rubin. 2009. Toward best practices in radiology reporting. *RADIOLOGY*, 252(3):852–856.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. 2023. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *Preprint*, arXiv:2306.00890.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Preprint*, arXiv:2304.08485.

Zhengliang Liu, Aoxiao Zhong, Yiwei Li, Longtao Yang, Chao Ju, Zihao Wu, Chong Ma, Peng Shu, Cheng Chen, Sekeun Kim, Haixing Dai, Lin Zhao, Lichao Sun, Dajiang Zhu, Jun Liu, Wei Liu, Dinggang Shen, Xiang Li, Quanzheng Li, and Tianming Liu. 2024. Radiology-gpt: A large language model for radiology. *Preprint*, arXiv:2306.08666.

Yasuhide Miura, Yuhao Zhang, Emily Tsai, Curtis Langlotz, and Dan Jurafsky. 2021. Improving factual completeness and consistency of image-to-text radiology report generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5288–5304, Online. Association for Computational Linguistics.

Maram Mahmoud A Monshi, Josiah Poon, and Vera Chung. 2020. Deep learning in generating radiology reports: A survey. *Artificial Intelligence in Medicine*, 106:101878.

Michael Moor, Qian Huang, Shirley Wu, Michihiro Yasunaga, Cyril Zakka, Yash Dalmia, Eduardo Pontes Reis, Pranav Rajpurkar, and Jure Leskovec. 2023. Med-flamingo: a multimodal medical few-shot learner. *Preprint*, arXiv:2307.15189.

Aaron Nicolson, Jason Dowling, and Bevan Koopman. 2023. Improving chest x-ray report generation by leveraging warm starting. *Artificial Intelligence in Medicine*, 144:102633.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav

Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Andreas S. Panayides, Amir Amini, Nenad D. Filipovic, Ashish Sharma, Sotirios A. Tsaftaris, Alistair Young, David Foran, Nhan Do, Spyretta Golemati, Tahsin Kurc, Kun Huang, Konstantina S. Nikita, Ben P. Veasey, Michalis Zervakis, Joel H. Saltz, and Constantinos S. Pattichis. 2020. Ai in medical imaging informatics: Current challenges and future directions. *IEEE Journal of Biomedical and Health Informatics*, 24(7):1837–1857.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Sang-Min Park and Young-Gab Kim. 2023. Visual language integration: A survey and open challenges. *Computer Science Review*, 48:100548.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *Preprint*, arXiv:2103.00020.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory optimizations toward training trillion parameter models. *Preprint*, arXiv:1910.02054.

Andrew B. Sellergren, Christina Chen, Zaid Nabulsi, Yuanzhen Li, Aaron Maschinot, Aaron Sarna, Jenny Huang, Charles Lau, Sreenivasa Raju Kalidindi, Mozziyar Etemadi, Florencia Garcia-Vicente, David Melnick, Yun Liu, Krish Eswaran, Daniel Tse, Neeral Beladia, Dilip Krishnan, and Shravya Shetty. 2022. Simplified transfer learning for chest radiography models using less data. *Radiology*, 305(2):454–465. PMID: 35852426.

Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Y. Ng, and Matthew P. Lungren. 2020. Chexbert: Combining automatic labelers and expert annotations for accurate radiology report labeling using bert. *Preprint*, arXiv:2004.09167.

Tim Tanida, Philip Müller, Georgios Kaissis, and Daniel Rueckert. 2023. Interactive and explainable region-guided radiology report generation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models.*, 3(6):7.

Tao Tu, Shekoofeh Azizi, Danny Driess, Mike Schaekermann, Mohamed Amin, Pi-Chuan Chang, Andrew Carroll, Chuck Lau, Ryutaro Tanno, Ira Ktena, Basil Mustafa, Aakanksha Chowdhery, Yun Liu, Simon Kornblith, David Fleet, Philip Mansfield, Sushant Prakash, Renee Wong, Sunny Virmani, Christopher Semturs, S Sara Mahdavi, Bradley Green, Ewa Dominowska, Blaise Aguera y Arcas, Joelle Barral, Dale Webster, Greg S. Corrado, Yossi Matias, Karan Singhal, Pete Florence, Alan Karthikesalingam, and Vivek Natarajan. 2023. Towards generalist biomedical ai. *Preprint*, arXiv:2307.14334.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Shawn Xu, Lin Yang, Christopher Kelly, Marcin Sieniek, Timo Kohlberger, Martin Ma, Wei-Hung Weng, Atilla Kiraly, Sahar Kazemzadeh, Zakkai Melamed, Jungyeon Park, Patricia Strachan, Yun Liu, Chuck Lau, Preeti Singh, Christina Chen, Mozziyar Etemadi, Sreenivasa Raju Kalidindi, Yossi Matias, Katherine Chou, Greg S. Corrado, Shravya Shetty, Daniel Tse, Shruthi Prabhakara, Daniel Golden, Rory Pilgrim, Krish Eswaran, and Andrew Sellergren. 2023. Elixr: Towards a general purpose x-ray artificial intelligence system through alignment of large language models and radiology vision encoders. *Preprint*, arXiv:2308.01317.

Benjamin Yan, Ruochen Liu, David E. Kuo, Subathra Adithan, Eduardo Pontes Reis, Stephen Kwak, Vasantha Kumar Venugopal, Chloe P. O'Connell, Agustina Saenz, Pranav Rajpurkar, and Michael Moor. 2023. Style-aware radiology report generation with radgraph and few-shot prompting. *Preprint*, arXiv:2310.17811.

Feiyang Yu, Mark Endo, Rayan Krishnan, Ian Pan, Andy Tsai, Eduardo Pontes Reis, Eduardo Kaiser Ururahy Nunes Fonseca, Henrique Min Ho Lee, Zahra Shakeri Hossein Abad, Andrew Y Ng, et al.

2023. Evaluating progress in automatic chest x-ray radiology report generation. *Patterns*, 4(9).

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

SICAR at RRG2024: GPU Poor's Guide to Radiology Report Generation

Kiartnarin Udomlapsakul[1,†], **Parinthapat Pengpun**[2,†], **Tossaporn Saengja**[3], **Kanyakorn Veerakanjana**[1,4,5]
Krittamate Tiankanon[1], **Pitikorn Khlaisamniang**[7], **Pasit Supholkhan**[7], **Amrest Chinkamol**[3]
Pubordee Aussavavirojekul[1], **Hirunkul Phimsiri**[1], **Tara Sripo**[7], **Chiraphat Boonnag**[7], **Trongtum Tongdee**[7]
Thanongchai Siriapisith[7], **Pairash Saiviroonporn**[7], **Jiramet Kinchagawat**[1,‡], **Piyalitt Ittichaiwong**[1,4,6,‡]

[1] PreceptorAI team, CARIVA Thailand, [2] Bangkok Christian International School,
[3] Vidyasirimedhi Institute of Science and Technology (VISTEC),
[4] Siriraj Informatics and Data Innovation Center (SIData+), Faculty of Medicine, Siriraj Hospital, Mahidol University,
[5] Social, Genetic and Developmental Psychiatry Centre,
Institute of Psychiatry, Psychology and Neuroscience, King's College London,
[6] School of Biomedical Engineering & Imaging Sciences, King College London,
[7] Radiology Department, Faculty of Medicine, Siriraj Hospital, Mahidol University
† Equal first contributions, ‡ Corresponding authors.

Abstract

Radiology report generation (RRG) aims to create free-text radiology reports from clinical imaging. Our solution employs a lightweight multimodal language model (MLLM) enhanced with a two-stage post-processing strategy, utilizing a Large Language Model (LLM) to boost diagnostic accuracy and ensure patient safety. We introduce the **"First, Do No Harm" SafetyNet**, which incorporates X-Raydar, an advanced X-ray classification model, to cross-verify the model outputs and specifically address false negative errors from the MLLM. This comprehensive approach combines the efficiency of lightweight models with the robustness of thorough post-processing techniques, offering a reliable solution for radiology report generation. Our system achieved fourth place on the F1-Radgraph metric for findings generation in the Radiology Report Generation Shared Task (RRG24).[1]

1 Introduction

Radiology is indispensable in healthcare, offering non-invasive methods to diagnose and monitor medical conditions. Central to this practice are radiology reports, which provide detailed interpretations of medical images crucial for clinical decision-making (Mityul et al., 2018). However, writing these reports is a meticulous process that demands significant domain expertise (Hartung et al., 2020). Radiologists must manually review images and formulate descriptive narratives, a task that is not only time-consuming but also susceptible to variability and errors, potentially affecting patient care and outcomes (Alexander et al., 2022).

One of the primary challenges in radiology report writing is the sheer volume of imaging studies that radiologists must interpret (Bruls and Kwee, 2020; Zhan et al., 2020). With the increasing use of imaging modalities such as computed tomography (CT), magnetic resonance imaging (MRI), and X-ray, radiologists are facing a growing workload that exceeds their capacity to provide timely and accurate reports (Winder et al., 2021; Bruls and Kwee, 2020). This challenge is further compounded by the rising demand for imaging services due to an aging population and the increasing prevalence of chronic diseases.

Another imperative issue in radiology report generation is the variability in report quality and consistency (Minn et al., 2015; Pool and Goergen, 2010). Different radiologists may interpret the same set of images differently, leading to inconsistencies in the information provided in the reports. This variability can stem from differences in writing styles, experience levels, and individual biases, all of which can have significant implications for patient care (Plumb et al., 2009; Naik et al., 2001; Brady et al., 2012). Inconsistencies in reports may lead to missed diagnoses or incorrect treatment decisions, underscoring the importance of standardized and automated approaches to report generation.

To address these challenges, Automated systems

[1] https://stanford-aimi.github.io/RRG24/

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 635–644
August 16, 2024. ©2024 Association for Computational Linguistics

have the potential to enhance the efficiency and accuracy of radiology report generation (Liao et al., 2023; Pang et al., 2023; Liu et al., 2023). These systems can reduce the time and effort required by radiologists while standardizing reporting practices to ensure consistency and relevance in reports. Moreover, automation can help address the increasing workload and demand for imaging services.

As Large Language Models (LLMs) have become widely available, numerous studies have explored the development of Multimodal LLMs (MLLMs) capable of natively processing additional modalities, such as images (Lu et al., 2023; Yang et al., 2023). Although there have been significant advancements in the development of MLLMs for various tasks (Chen et al., 2023; Wu et al., 2023), none have specifically focused on lightweight models for the medical domain.

Local deployment is critical as many hospitals are concerned that uploading images to the cloud for AI processing may violate privacy laws such as the General Data Protection Regulation (GDPR) in Europe or the Personal Data Protection Act (PDPA) in Thailand. Addressing this issue is essential to ensure that patients can receive enhanced medical services while maintaining their privacy. Additionally, most hospitals in developing countries are GPU-constrained and lack access to high-end GPUs which are typically required for deployment. Therefore, it is imperative to develop lightweight models capable of performing inference on-premise using consumer-grade GPUs.

Motivated by these challenges, we investigate various architectures with a focus on identifying models that offer the optimal cost-to-performance ratio for local deployment. For the purposes of this study, we concentrate on the task of findings generation.

Our contributions are summarized as follows:

- We developed and trained a lightweight Multimodal Large Language Model (MLLM) for the radiology report generation task using a two-stage training strategy, achieving performance metrics comparable to those of larger models.
- We introduced a novel two-stage post-processing strategy. The first stage enhances the readability and clarity of the reports. The second stage, "First, Do No Harm" SafetyNet, employs the X-Raydar classification model to cross-verify the model outputs, significantly improving diagnostic accuracy and ensuring patient safety.

2 Methodology

2.1 Model Architecture

Impressed by its superior performance, which surpasses even some larger models despite its lightweight nature in general domain, we decided to follow model architecture design of Bunny for this study (He et al., 2024). Our model components include the SigLIP-so400m[2] (Zhai et al., 2023) as the visual encoder, a two-layer Multi-layer perceptron (MLP) with a GELU activation as the vision-language connector, and the Phi-2 2.7B as our LLM (Hughes, 2023).

The SigLIP visual encoder extracts meaningful features from chest X-ray images, enabling the model to capture relevant visual information. The MLP integrates these visual features with language representations. Phi-2, a 2.7 billion parameter lightweight language model trained on high-quality data, achieves performance metrics comparable to substantially larger models. It demonstrates exceptional proficiency in benchmarks such as commonsense reasoning, language comprehension, question-answering, and coding tasks, frequently surpassing models with significantly more parameters.

2.2 Training Strategy & Datasets

We employ a two-stage training strategy to optimize our model's performance. In the first stage, we train only the MLP connector using the LLaVa-Med alignment 500k dataset (Li et al., 2023; Zhang et al., 2023), while keeping the rest of the model frozen. LLaVa-Med is a large-scale dataset specifically curated for medical vision-language tasks, containing a diverse collection of medical imaging modalities and tasks. By pretraining on this dataset, the MLP connector learns to effectively map visual features to language representations in the medical domain.

The second stage involves fine-tuning both the vision-language connector and the Language Model (LLM), while keeping the visual encoder frozen. This fine-tuning process utilizes the interpret-cxr dataset (Xu et al., 2024) comprising a mixture of multiple chest X-ray datasets: CheXpert (Chambon et al., 2024), PadChest (Bustos et al., 2020), BIMCV COVID-19 (Vayá et al., 2020), and MIMIC-CXR-JPG (Johnson et al., 2019). This

[2] SigLIP HuggingFace Link

dataset includes chest X-ray images along with their corresponding radiology reports, providing task-specific training data. For our study, we combined both findings and impressions into a single dataset, totaling 710,807 image-text pairs. In our preliminary study, retaining only the first image from each study outperformed using all images, as shown in 3. Therefore, we heuristically kept only the first image to preserve report diversity.

2.3 Two-Stage Post-Processing Strategy

In addition to our model's architecture and training strategies, we implement a crucial post-processing strategy, wherein the model outputs undergo sequential processing to enhance the overall quality of the reports (See Appendix C for detailed prompts).

2.3.1 First stage: Report Refinement

In the first stage, we utilize a Large Language Model (LLM) to enhance the comprehensiveness of the findings reports. Our key objectives are to improve readability and clarity, eliminate nonsensical words, and remove duplicated sentences from the model hallucinations. For normal chest X-ray (CXR) findings, we provide detailed, standardized explanations to clarify the condition. We use a recommended vocabulary list to maintain consistency across reports. Our methodology promotes concise reporting by focusing on critical findings, while still adhering to a professional radiology report format. This includes transforming simple statements like "No significant findings" into comprehensive and detailed descriptions.

2.3.2 Second stage: "First, Do No Harm" SafetyNet

This post-processing strategy, termed "First, Do No Harm" SafetyNet, involves using an advanced X-ray classification model, X-Raydar, to provide a second opinion on chest X-ray images. This methodology mirrors the practice of doctors consulting with colleagues to validate the diagnoses, thereby mitigating the risk of errors that could potentially harm patients.

X-Raydar Integration X-Raydar, a state-of-the-art X-ray classification model, is trained on a substantial dataset of 1.8 million chest X-rays, covering a wide range of pathologies (Cid et al., 2024). By integrating X-Raydar into our post-processing strategy, we leverage its robust performance to cross-verify and refine the outputs generated by our MLLM.

Second Opinion Inference A major challenge in findings generation is the occurrence of false negative errors, such as incorrectly reporting "lungs are clear" or "no cardiomegaly" To mitigate this issue, we use Llama3 70B[3] with a specially designed prompt to detect and correct such critical errors. The prompt incorporates the classification results from X-Raydar to specifically address common false negative errors. For example, if X-Raydar identifies signs of cardiomegaly but the initial report states "no cardiomegaly," our tailored prompt for Llama3 ensures that the final report accurately reflects the patient's condition. This dual-check strategy significantly increases agreement with the ground truth report, thereby improving diagnostic accuracy and enhancing patient safety.

3 Experimental Setup

3.1 Evaluation

We evaluated our approach using metrics for natural language generation (NLG) quality and clinical accuracy, as implemented by the Vilmedic framework (Delbrouck et al., 2022b).

NLG Metrics

- BLEU measures the precision of n-grams in the generated text compared to a reference text (Papineni et al., 2002).
- ROUGE-L focuses on the longest common subsequence between the generated and reference texts (Lin, 2004).
- BERTscore uses contextual embeddings to compare semantic similarity between the generated and reference texts (Zhang et al., 2019).

Clinical Accuracy Metrics

- F1-CheXbert computes the F1 score based on the similarity of indicator vectors for 14 pathologies (Smit et al., 2020).
- F1-RadGraph calculates the overlap in clinical entities and relations extracted from the reports (Delbrouck et al., 2022a).

These metrics provide a comprehensive evaluation of our model's performance in generating accurate and clinically relevant radiology reports.

3.2 Model Architecture Ablations

To investigate the complex relationship between model architecture and overall performance across

[3] LLaMa3 70B Instruct HuggingFace Link

Table 1: Performance of Various LLM and Visual Encoder Combinations on the Public Findings Benchmark (One Epoch ≈ 5000 Steps)

Model	Step	BLEU4	ROUGEL	Bertscore	F1-cheXbert	F1-RadGraph
Phi-2 + SigLIP	4000	5.83	20.98	46.72	49.69	19.21
Phi-2 + SigLIP	8000	6.93	**23.41**	50.81	**55.70**	22.05
Phi-2 + SigLIP	12000	6.96	23.26	51.63	52.91	**22.86**
Phi-2 + SigLIP (S2)	4000	5.08	19.85	45.67	47.96	18.53
Phi-2 + SigLIP (S2)	8000	**7.47**	23.38	50.56	55.30	22.32
Phi-2 + SigLIP (S2)	12000	7.3	22.9	**50.82**	52.84	21.93
Llama3 (OpenBio) + SigLIP (S2)	4000	2.26	16.03	41.42	37.49	12.98
Llama3 (OpenBio) + SigLIP (S2)	8000	5.21	20.32	47.06	45.78	18.11
Llama3 (OpenBio) + SigLIP (S2)	12000	6.01	20.75	48.24	49.72	18.09

various metrics and tasks, we designed and conducted the following series of experiments to isolate specific architectural elements and their effects:

- Language Model:
  - Phi-2 2.7B
  - Llama3-OpenBioLLM 8B[4]
- Visual Encoder:
  - SigLIP
  - SigLIP with S2-Wrapper

In addition to our base model, Phi-2 2.7B with the SigLIP visual encoder, we conducted further ablation studies to understand the impact of different model architectures. For the language model (LLM), we selected Llama3-OpenBioLLM 8B as our larger model to investigate whether initializing from a medical LLM could enhance the performance of a MLLM on findings generation task. The model was fine-tuned using a comprehensive dataset of high-quality biomedical data, allowing it to comprehend and generate text with precise domain-specific accuracy and fluency. The Llama3-OpenBioLLM 8B demonstrated exceptional performance on multiple medical LLM benchmarks[5], surpassing even some larger models.

For the visual encoder, we employed the S2-Wrapper, an extension designed to extract multi-scale features from images (Shi et al., 2024). This approach was chosen to evaluate the impact of multi-scale feature extraction on the findings generation task. The integration of the S2-Wrapper aims to enhance the model's ability to handle complex visual features and improve the overall accuracy of the generated reports.

4 Results & Discussion

4.1 Model Architecture Ablations

Our best architecture, Phi-2 combined with SigLIP visual encoder, demonstrates superior performance as indicated by the F1-Radgraph metric as presented in Table 1. Notably, this configuration outperforms the S2-wrapper extension. We hypothesize that the general domain SigLIP visual encoder encounters difficulties in effectively extracting useful information from X-ray images at multiple scales. Additionally, this architecture surpasses the performance of the larger medical domain Llama3-OpenBioLLM 8B, suggesting that the success in this specific findings generation task may be more dependent on the quality of image information extracted by the visual encoder rather than the pretrained knowledge of LLMs.

4.2 Post-processing

Table 2: Performance improvement of each post-processing stage on Hidden Findings Benchmark.

Model	F1-RadGraph
Phi-2 + SigLIP	22.61
Phi-2 + SigLIP (Stage 1)	23.11 (+0.5)
Phi-2 + SigLIP (Stage 1&2)	24.62 (+1.51)

Our two-stage post-processing strategy markedly improves the performance metrics for our findings generation task, as demonstrated by the hidden-findings test results in Table 2. In the first stage, report refinement increased the F1-Radgraph metric from 22.61 to 23.11 (+0.5). The incorporation of the "First, Do No Harm" SafetyNet in the second stage further elevated the F1-Radgraph metric from 23.11 to 24.62 (+1.51), resulting in a total improvement of 2.01 points over the default model. This comprehensive approach not only enhances report readability but also significantly boosts diagnostic accuracy and patient safety, leading to higher quality radiology reports.

5 Conclusion

We present our approach to the Radiology Report Generation task in the BioNLP 2024 shared task. This study investigates various training configurations and data mixtures to develop lightweight models for generating radiology reports from chest X-ray images. Our findings demonstrate that even a smaller model, such as the Phi-2 language model, can perform comparably to larger models in the

[4] LLaMa3 OpenBioLLM 8B HuggingFace Link

[5] OpenLLM Leaderboard

report generation task. Additionally, incorporating post-processing techniques significantly enhances the quality of the reports and ensures patient safety. This is particularly crucial for hospitals in resource-constrained settings. By focusing on models that can be fine-tuned on a single A100 GPU and operated on-premises with a consumer-grade GPU, we address privacy concerns and improve the accessibility of this technology.

Limitations

In this work, we utilized the Llama3-70b-instruct model on HuggingChat for post-processing in both stages, demonstrating that it improves the metric (F1-RadGraph) of the generated reports. However, we did not explicitly analyze the quality of post-processing with smaller LLMs to determine if they can achieve similar results. Future research could explore post-processing with multiple LLM sizes to understand the impact of model size on performance. Additionally, our current approach involves sequential two-stage post-processing, which may not fully leverage the LLM's capabilities and could introduce unnecessary complexity and latency. Combining these stages into a single step could reduce latency and streamline the overall process.

Acknowledgments

This research is supported by PreceptorAI team and CARIVA Thailand. The experiments were conducted with the resources and services provided by the Siriraj Radiology Department (SIRAD). Their resources and domain expertise were instrumental in the successful completion of this study.

References

Robert Alexander, Stephen Waite, Michael A Bruno, Elizabeth A Krupinski, Leonard Berlin, Stephen Macknik, and Susana Martinez-Conde. 2022. Mandating limits on workload, duty, and speed in radiology. *Radiology*, 304(2):274–282.

Adrian P. Brady, Risteárd Ó Laoide, Peter A McCarthy, and Ronan McDermott. 2012. Discrepancy and error in radiology: Concepts, causes and consequences. *The Ulster Medical Journal*, 81:3 – 9.

RJM Bruls and RM Kwee. 2020. Workload for radiologists during on-call hours: dramatic increase in the past 15 years. *Insights into imaging*, 11:1–7.

Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria De La Iglesia-Vaya. 2020. Padchest: A large chest x-ray image dataset with multi-label annotated reports. *Medical image analysis*, 66:101797.

Pierre Chambon, Jean-Benoit Delbrouck, Thomas Sounack, Shih-Cheng Huang, Zhihong Chen, Maya Varma, Steven QH Truong, Chu The Chuong, and Curtis P. Langlotz. 2024. Chexpert plus: Hundreds of thousands of aligned radiology texts, images and patients.

Ke Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. 2023. Shikra: Unleashing multimodal llm's referential dialogue magic. *ArXiv*, abs/2306.15195.

Zhihong Chen, Maya Varma, Jean-Benoit Delbrouck, Magdalini Paschali, Louis Blankemeier, Dave Van Veen, Jeya Maria Jose Valanarasu, Alaa Youssef, Joseph Paul Cohen, Eduardo Pontes Reis, Emily B. Tsai, Andrew Johnston, Cameron Olsen, Tanishq Mathew Abraham, Sergios Gatidis, Akshay S. Chaudhari, and Curtis Langlotz. 2024. Chexagent: Towards a foundation model for chest x-ray interpretation. *Preprint*, arXiv:2401.12208.

Yashin Dicente Cid, Matthew Macpherson, Louise Gervais-Andre, Yuanyi Zhu, Giuseppe Franco, Ruggiero Santeramo, Chee Lim, Ian Selby, Keerthini Muthuswamy, Ashik Amlani, et al. 2024. Development and validation of open-source deep neural networks for comprehensive chest x-ray reading: a retrospective, multicentre study. *The Lancet Digital Health*, 6(1):e44–e57.

Jean-Benoit Delbrouck, Pierre Chambon, Christian Bluethgen, Emily Tsai, Omar Almusa, and Curtis Langlotz. 2022a. Improving the factual correctness of radiology report generation with semantic rewards. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4348–4360.

Jean-benoit Delbrouck, Khaled Saab, Maya Varma, Sabri Eyuboglu, Pierre Chambon, Jared Dunnmon, Juan Zambrano, Akshay Chaudhari, and Curtis Langlotz. 2022b. Vilmedic: a framework for research at the intersection of vision and language in medical ai. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 23–34.

Michael P Hartung, Ian C Bickle, Frank Gaillard, and Jeffrey P Kanne. 2020. How to create a great radiology report. *Radiographics*, 40(6):1658–1670.

Muyang He, Yexin Liu, Boya Wu, Jianhao Yuan, Yueze Wang, Tiejun Huang, and Bo Zhao. 2024. Efficient multimodal learning from data-centric perspective. *arXiv preprint arXiv:2402.11530*.

Alyssa Hughes. 2023. Phi-2: The surprising power of small language models — microsoft.com. [Accessed 18-05-2024].

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng.

2019. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. 2023. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *arXiv preprint arXiv:2306.00890*.

Yuxiang Liao, Hantao Liu, and Irena Spasic. 2023. Deep learning approaches to automatic radiology report generation: A systematic review. *Informatics in Medicine Unlocked*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Chang Liu, Yuanhe Tian, and Yan Song. 2023. A systematic review of deep learning-based research on radiology report generation. *ArXiv*, abs/2311.14199.

Yuzhe Lu, Sungmin Hong, Yash Shah, and Panpan Xu. 2023. Effectively fine-tune to improve large multimodal models for radiology report generation. *ArXiv*, abs/2312.01504.

Matthew J Minn, Arash R Zandieh, and Ross W Filice. 2015. Improving radiology report quality by rapidly notifying radiologist of report errors. *Journal of digital imaging*, 28:492–498.

Marina I Mityul, Brian Gilcrease-Garcia, Mark D Mangano, Jennifer L Demertzis, and Andrew J Gunn. 2018. Radiology reporting: current practices and an introduction to patient-centered opportunities for improvement. *American Journal of Roentgenology*, 210(2):376–385.

Sandeep S Naik, Anthony Hanbidge, and Stephanie R Wilson. 2001. Radiology reports: examining radiologist and clinician preferences regarding style and content. *American Journal of Roentgenology*, 176(3):591–598.

Ting Pang, Peigao Li, and Lijie Zhao. 2023. A survey on automatic generation of medical imaging reports based on deep learning. *BioMedical Engineering OnLine*, 22.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

AAO Plumb, FM Grieve, and SH Khan. 2009. Survey of hospital clinicians' preferences regarding the format of radiology reports. *Clinical radiology*, 64(4):386–394.

Felicity Pool and Stacy Goergen. 2010. Quality of the written radiology report: a review of the literature. *Journal of the American College of Radiology*, 7(8):634–643.

Baifeng Shi, Ziyang Wu, Maolin Mao, Xin Wang, and Trevor Darrell. 2024. When do we not need larger vision models? *arXiv preprint arXiv:2403.13043*.

Akshay Smit, Saahil Jain, Pranav Rajpurkar, Anuj Pareek, Andrew Y Ng, and Matthew Lungren. 2020. Combining automatic labelers and expert annotations for accurate radiology report labeling using bert. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1500–1519.

Maria De La Iglesia Vayá, Jose Manuel Saborit, Joaquim Angel Montell, Antonio Pertusa, Aurelia Bustos, Miguel Cazorla, Joaquin Galant, Xavier Barber, Domingo Orozco-Beltrán, Francisco García-García, et al. 2020. Bimcv covid-19+: a large annotated dataset of rx and ct images from covid-19 patients. *arXiv preprint arXiv:2006.01174*.

Mateusz Winder, Aleksander Jerzy Owczarek, Jerzy Chudek, Joanna Pilch-Kowalczyk, and Jan Baron. 2021. Are we overdoing it? changes in diagnostic imaging workload during the years 2010–2020 including the impact of the sars-cov-2 pandemic. In *Healthcare*, volume 9, page 1557. MDPI.

Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. 2023. Next-gpt: Any-to-any multimodal llm. *ArXiv*, abs/2309.05519.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Ling Yang, Zhanyu Wang, Zhenghao Chen, Xinyu Liang, and Luping Zhou. 2023. Medxchat: A unified multimodal large language model framework towards cxrs understanding and generation.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. *Preprint*, arXiv:2303.15343.

Henry Zhan, Kevin M. Schartz, Matthew E. Zygmont, Jamlik-Omari Johnson, and Elizabeth A. Krupinski. 2020. The impact of fatigue on complex ct case interpretation by radiology residents. *Academic radiology*.

Sheng Zhang, Yanbo Xu, Naoto Usuyama, Hanwen Xu, Jaspreet Bagga, Robert Tinn, Sam Preston, Rajesh Rao, Mu Wei, Naveen Valluri, et al. 2023. Biomedclip: a multimodal biomedical foundation model pretrained from fifteen million scientific image-text pairs. *arXiv preprint arXiv:2303.00915*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Preliminary Study

To investigate the impact of data composition and transfer learning on model performance, we conducted the following experiments:

A.1 Initialization and Training Strategies:

We explored two initialization and training strategies:

- Initialization from MLLM Pretrained on General Domain: This approach involves continually fine-tuning the pretrained multimodal language model (MLLM) on the interpret-cxr dataset, focusing solely on the final stage of training (stage 2).
- Initialization from LLM and Randomly Initialized Adapter: In this method, the language model (LLM) is pretrained on the LLaVa-MED dataset (stage 1) with a randomly initialized adapter, followed by fine-tuning on the interpret-cxr dataset (stage 2). This two-stage process aims to leverage domain-specific pretraining to enhance performance.

Our results indicated that this two-stage approach, which leverages domain-specific knowledge from LLaVa-MED, is beneficial and enhances performance.

A.2 Image Selection

We examined the effect of different image selection techniques:

- Reusing the Same Report When Multiple Images Are Provided: This technique involves using all available images for a given report, resulting in 1 million image-text pairs. This approach aims to maximize the amount of visual information provided to the model.
- Using Only the First X-ray Image When Multiple Images Are Provided: Here, only the first image from each study is used, leading to a dataset of 700,000 image-text pairs. This method is intended to reduce redundancy and potential bias in the reports by focusing on the most relevant image.

Our data mixture study revealed that using only the first image from each study yielded slightly better performance than using all images, ensuring the diversity of the radiology reports.

B Dataset Cleaning

In the preliminary inspection of the dataset, we observed that numerous reports within the interpret-cxr dataset contained sentences with information that could not be derived solely from the X-ray images. These sentences included details such as dates, doctor information, references to other imaging modalities, and comparisons with previous findings. Such extraneous information introduces noise that may lead the model to hallucinate incorrect dates, numbers, and comparisons with non-existent prior studies (Chen et al., 2024).

To mitigate this issue, we attempted to utilize GPT-3.5 Turbo to remove this irrelevant information from the dataset. The dataset cleaning prompts and examples are detailed in Appendix B.1 and B.2. However, during the evaluation, we observed a slight decline in performance metrics, as illustrated in Table 4, following the removal of these sentences. We suspected that the public-test and hidden-test datasets did not undergo similar cleaning procedures, resulting in uncleaned test sets. Therefore, to maximize of our performance metrics, we decided to use the original dataset without data cleaning for the remaining of our study.

B.1 Cleaning Prompt

We provide the prompt used for preprocessing and cleaning the training dataset to remove information that cannot be obtained solely from X-ray images.

Findings: "Remove non-x-rays discernible information from chest x-ray findings i.e. date, previous report mentions and comparison, and information from other imaging modality. Keep all remaining sentences unchanged:"

Impression: "Remove non-x-rays discernible information from chest x-ray impression i.e. date, doctor information, previous report mentions and comparison, and information from other imaging modality. Keep all remaining sentences unchanged. But if there is nothing left, return |None| and stop generating:"

B.2 Examples

B.2.1 Comparison with previous report

Original: Compared with the previous one, the x-ray is slightly inspired. no lung consolidations or pleural effusion are observed.

Clean: No lung consolidations or pleural effusion are observed.

B.2.2 Date Mentions

Original: AP chest radiograph on 12/11/08 at 2315 demonstrates a dual lead AICD. Stable cardiomegaly and stable left basilar opacities, likely

Table 3: Preliminary Evaluation of Initialization Strategies and Image Selection for Radiology Report Generation. This table compares the performance metrics of models initialized from a pretrained MLLM versus those initialized from an LLM with a randomly initialized adapter, as well as the impact of using only the first image from each study versus using all provided images. The results indicate that initializing from an LLM with a randomly initialized adapter yields better performance, and selecting the first image from each study slightly improves the metrics. Consequently, we heuristically retained only the first image to reduce redundancy and maintain report diversity.

LLM + Visual Encoder	Train Data	Epoch	BLEU4	ROUGEL	Bertscore	F1-cheXbert	F1-RadGraph
Phi-2 + SigLIP (init from Bunny)	LLaVa-Med + CXR	1	4.84	20.05	45.81	48.39	18.13
Phi-2 + SigLIP	LLaVa-Med + CXR	1	5.74	20.72	47.32	49.46	19.13
Phi-2 + SigLIP	LLaVa-Med + CXR (First)	1	5.45	21.17	47.43	51.10	19.52

Table 4: Results of the dataset cleaning experiment on findings and impressions. We performed stage 2 finetuning on the SigLIP and Phi-2 2.7B model architecture with different data mixtures for this experiment. "Raw" refers to the original interpret-cxr dataset, while "Clean" denotes the dataset cleaned by GPT-3.5 Turbo using the specified cleaning prompt.

Report Type	Train Data	Epoch	BLEU4	ROUGEL	Bertscore	F1-cheXbert	F1-RadGraph
findings	Raw	1	6.19	24.49	47.61	43.91	18.08
findings	Clean	1	5.84	24.17	47.14	44.19	17.83
impression	Raw	1	9.87	27.65	50.57	51.80	23.96
impression	Clean	1	6.62	23.66	50.74	49	24.62

atelectasis. Persistent right-sided pleural effusion. Diffuse reticular opacities, mild interstitial edema. Elevation of the left hemidiaphragm. Hiatal hernia. Partially visualized abdominal aortic stent graft. AP chest radiograph on 12-11-2008 at 3:11 a.m. demonstrates no significant interval change in cardiopulmonary status.

Clean: AP chest radiograph demonstrates a dual lead AICD. Stable cardiomegaly and stable left basilar opacities, likely atelectasis. Persistent right-sided pleural effusion. Diffuse reticular opacities, mild interstitial edema. Elevation of the left hemidiaphragm. Hiatal hernia. Demonstrates no significant interval change in cardiopulmonary status.

B.2.3 Other Modalities Mentions

Original: Chest x-ray. bilateral bronchiectasis with a predominance on the right side, noting an increase in density around these right basal bronchiectasis in relation to consolidations described in previous ct. there is no pleural effusion. cardiomedastinal silhouette and hila are within normal limits. biapical caps. bone and soft parts without notable findings.

Clean: Chest x-ray. Bilateral bronchiectasis with a predominance on the right side. There is no pleural effusion. Cardiomedastinal silhouette and hila are within normal limits. Biapical caps. Bone and soft parts without notable findings.

B.2.4 Doctor information

Original: 1.Interval development and resolution of a right upper lobe opacification, possibly representing interval resolution of right upper lobe aspiration or asymmetric pulmonary edema. 2. Persistent small bilateral pleural effusions. ""Physician to Physician Radiology Consult Line: (753) 619-1110"" I have personally reviewed the images for this examination and agreed with the report transcribed above.

Clean: 1.Interval development and resolution of a right upper lobe opacification, possibly representing interval resolution of right upper lobe aspiration or asymmetric pulmonary edema. 2. Persistent small bilateral pleural effusions."

C LLM Prompts

We provide a template of our post-processing prompt for the LLM to enhance diverse aspects generated report. The Report Refinement prompt enhance the readability and clarify the report while the "First, Do No Harm" SafetyNet prompt of Llama3 combines the results of our MLLM model and the classification results from X-Raydar.

Prompt 1: First Stage: Report Refinement

Radiology Reporting Instructions

You are an experienced radiologist tasked with interpreting CXR images and generating reports from free-text descriptions. Your primary objectives are to:

- Enhance readability and clarity of the text.
- Conduct Radgraph sterilization to ensure data integrity and accuracy.

When processing normal CXR findings, provide detailed explanations to clarify the condition. For instance:

Input Examples

- No significant findings.
- No acute cardiopulmonary findings.
- No acute cardiopulmonary abnormality.
- The heart is normal in size. The mediastinum is unremarkable. The lungs are clear.
- The heart size and pulmonary vascularity appear within normal limits. The lungs are free of focal airspace disease. No pleural effusion or pneumothorax is seen.
- No acute cardiopulmonary findings.

Expected Output

- The lungs are clear. No cardiomegaly. The cardiomediastinal and hilar contours are normal. There is no focal consolidation, pleural effusion, or pneumothorax. The pulmonary vascular markings are normal. No free air beneath the diaphragm.

Use a recommended vocabulary list to standardize report language and maintain consistency across reports. This list includes [’AC’, ’Bony’, ’Borderline’, ’CHF’, ’Calcified’, ’Cardiac’, ’Cardiomediastinal’, ’Cardiomegaly’, ’Clips’, ’Dense’, ’Dobbhoff’, ’Esophageal’, ’Extensive’, ’Heart’, ’Healing’, ’Hilar’, ’Hyperinflated’, ’IJ’, ’Increase’, ’Increased’, ’Interval’, ’Interposition’, ’Lung’, ’Lungs’, ’Lucency’, ’Minimal’, ’Moderate’, ’Mild’, ’Mildly’, ’Monitoring’, ’Multiple’, ’Nasogastric’, ’Nearly’, ’New’, ’Normal’, ’Orphaned’, ’PICC’, ’Pneumomediastinum’, ’Pneumothorax’, ’Port - A - Cath’, ’Pulmonary’, ’Right-sided’, ’Small’, ’Slight’, ’Slightly’, ’Stable’, ’Subcutaneous’, ’Tip’, ’Venous’, ’Widespread’, ’Worsening’, ’Zone’, ’accessory’, ’acute’, ’adenocarcinoma’, ’air’, ’air-filled’, ’airspace’, ’along’, ’angles’, ’anterior’, ’anteriorly’, ’apparent’, ’appearance’, ’appropriately’, ’area’, ’areation’, ’artifact’, ’atelectasis’, ’axilla’, ’benign’, ’bibasal’, ’bilaterally’, ’blunting’, ’borderline’, ’bowel’, ’bronchovascular’, ’caliber’, ’calcification’, ’calcified’, ’cancer’, ’cardiac’, ’cardiomegaly’, ’central’, ’change’, ’chest’, ’chf’, ’clavicle’, ’clavicular’, ’clear’, ’clips’]
Reports should be styled succinctly, focusing on critical findings and summarizing significant observations without omitting essential details. Each report should follow the professional radiology report format:

Example of Good Reports

- The lungs are clear without focal consolidation. No pleural effusion or pneumothorax is seen. Heart size is top-normal. The mediastinal silhouette is unremarkable.
- Portable frontal radiograph of the chest demonstrates a right chest tube in unchanged position ending at the right apex. The right basilar pneumothorax continues to decrease in size. The pneumomediastinum is also decreasing. Extensive subcutaneous emphysema persists. Stable heart size and mediastinal contours. Small left pleural effusion is unchanged.
- The cardiac, mediastinal and hilar contours appear stable. Streaky left basilar opacity suggests minor atelectasis. The lateral view depicts a greater degree of right middle lobe atelectasis than before, more coalescent. There is no definite pleural effusion or pneumothorax.
- Persistent hila with a congestive appearance possibly due to pulmonary edema, but without evidence of significant consolidations or pleural effusion. to be correlated clinically.
- Cardiac silhouette is unchanged. Aortic arch calcification seen. Pulmonary vascularity is within normal limits. There is trace right pleural effusion noted. Bibasilar atelectasis is seen. There is no pneumothorax. Multilevel degenerative changes seen in the thoracic spine.

  (answer only summarize report to text paragraph)

Prompt 2: "Second Stage: First, Do No Harm" SafetyNet

Refine ’Input’ with ’Refine information’ indicating that the patient has conditions as refine information with the following conditions:

1. If the input and refined information have mismatched information, such as Refine information indicating an additional pathology not mentioned in the input, prioritize the refined information.

2. However, if the ’Refine information’ suggests a pathology already included in the ’Input’, we will not refine the input.

3. We will remove the sentence "lungs are clear" if there is any abnormality in the lung, pulmonary, or pleura.

Example 1:
Input = The lungs are clear. No cardiomegaly. The cardiomediastinal and hilar contours are normal. There is no focal consolidation, pleural effusion, or pneumothorax. The pulmonary vascular markings are normal.
Refine information = This patient has cardiomegaly and pleural effusion.
Output should be = There is cardiomegaly. The cardiomediastinal and hilar contours are normal. There is pleural effusion. There is no focal consolidation or pneumothorax. The pulmonary vascular markings are normal.
Example 2:
Input = The lungs are clear. No cardiomegaly. The cardiomediastinal and hilar contours are normal. There is no focal consolidation, pleural effusion, or pneumothorax. The pulmonary vascular markings are normal.
Refine information = This patient has cardiomegaly.
Output should be = There is cardiomegaly. The lungs are clear. The cardiomediastinal and hilar contours are normal. There is no focal consolidation, pleural effusion, or pneumothorax. The pulmonary vascular markings are normal.
Your answer should provide only the ’Output’ format and not include any other comments.

Shimo Lab at "Discharge Me!": Discharge Summarization by Prompt-Driven Concatenation of Electronic Health Record Sections

Yunzhen He[*1] **Hiroaki Yamagiwa**[*1] **Hidetoshi Shimodaira**[1,2]
[1] Kyoto University [2] RIKEN AIP
`he.yunzhen.25d@st.kyoto-u.ac.jp,`
`hiroaki.yamagiwa@sys.i.kyoto-u.ac.jp,shimo@i.kyoto-u.ac.jp`

Abstract

In this paper, we present our approach to the shared task "Discharge Me!" at the BioNLP Workshop 2024. The primary goal of this task is to reduce the time and effort clinicians spend on writing detailed notes in the electronic health record (EHR). Participants develop a pipeline to generate the "Brief Hospital Course" and "Discharge Instructions" sections from the EHR. Our approach involves a first step of extracting the relevant sections from the EHR. We then add explanatory prompts to these sections and concatenate them with separate tokens to create the input text. To train a text generation model, we perform LoRA fine-tuning on the ClinicalT5-large model. On the final test data, our approach achieved a ROUGE-1 score of 0.394, which is comparable to the top solutions.

1 Introduction

Electronic health records (EHR) eliminate the need for end-users to write medical records by hand and provide easy access to digital records (Menachemi and Collum, 2011). However, the use of EHR sometimes increases the burden on end-users (Shanafelt et al., 2016; Liu et al., 2022; Gao et al., 2023). With this in mind, there has been active research in recent years into applying natural language processing (NLP) to EHR to reduce the burden on end-users (Dong et al., 2022; Houssein et al., 2023; Veen et al., 2023).

To explore the potential of NLP in EHR, the shared task "Discharge Me!" (Xu et al., 2024) at the BioNLP Workshop 2024 evaluates the ability to generate discharge summaries. The goal of this task is to reduce the time and effort clinicians spend on writing detailed notes in the EHR. Participants develop a pipeline that leverages the EHR data to generate discharge summaries.

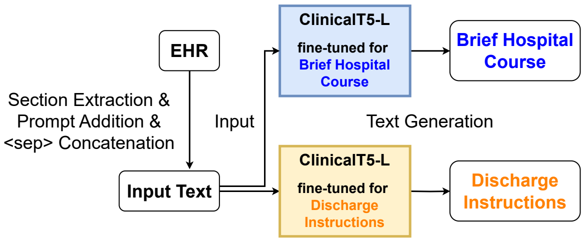


Figure 1: Overview of our pipeline. To create input text, we extract sections from the EHR, add explanatory prompts, and then concatenate them with `<sep>` tokens. We then generate discharge summaries using ClnicalT5-large, which has been fine-tuned for each target.

In this paper, we present our approach to the shared task. Fig. 1 provides an overview of our pipeline. We preprocess the EHR, as illustrated in Fig. 2, by removing noise and extracting sections that are essential for the target summary. The sections are selected based on a predetermined priority. For extracted sections, we prepend the prompt from Table 2 to the beginning of the text, concatenate these sections using `<sep>` tokens, and thus prepare the input text. We also removed noise from the target text. We then fine-tuned ClinicalT5 (Lu et al., 2022), which is pre-trained on clinical texts. On the final test data, our approach achieved a ROUGE-1 score of 0.394, which is comparable to the top solutions.

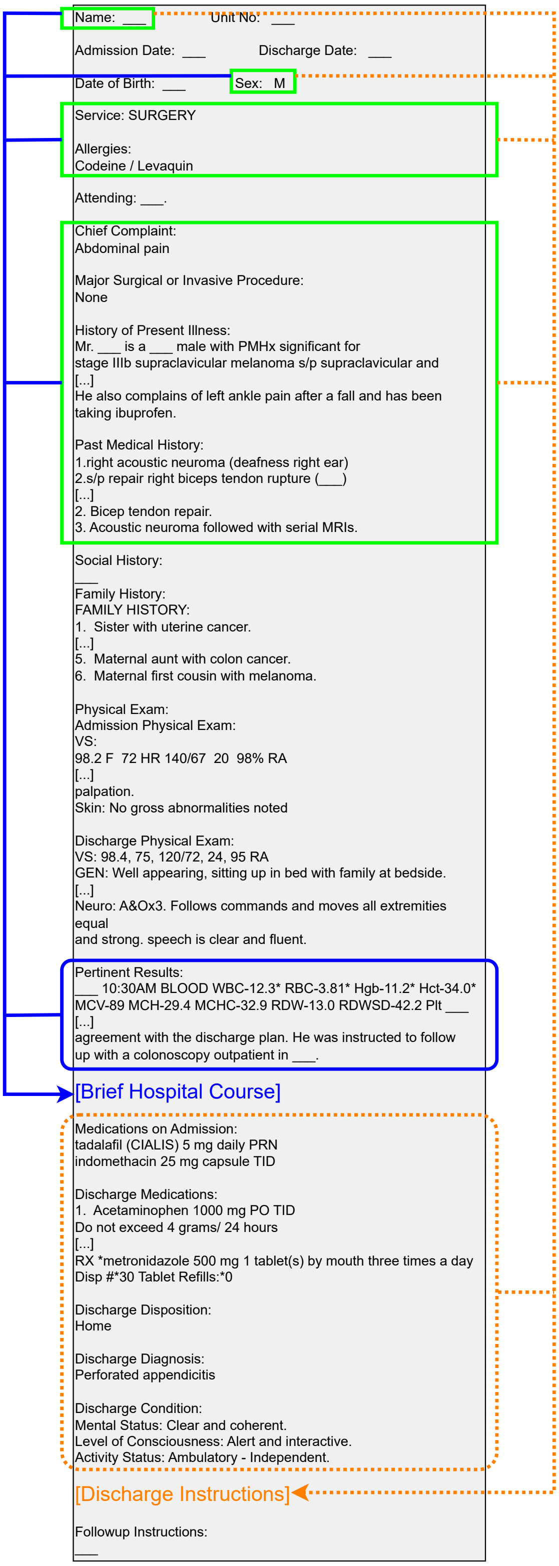
2 Related work

2.1 Text generation models in clinical domain

Decoder. ClinicalGPT (Wang et al., 2023), whose base model is BLOOM-7B (Le Scao et al., 2022), uses LoRA (Hu et al., 2022) for fine-tuning and applies the reinforcement learning process used in InstructGPT (Ouyang et al., 2022). BioMistral-7B (Labrak et al., 2024) underwent additional pretraining of the Mistral-7B (Jiang et al., 2023) model on PubMed Central (Roberts, 2001) and showed good performance on the clinical knowledge QA

[*] The first two authors contributed equally to this work. Our code is available at `https://github.com/githubhyz/DischargeMe_BioNLP2024`.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 645–657
August 16, 2024. ©2024 Association for Computational Linguistics



Name: ___ Unit No: ___

Admission Date: ___ Discharge Date: ___

Date of Birth: ___ Sex: M

Service: SURGERY

Allergies:
Codeine / Levaquin

Attending: ___.

Chief Complaint:
Abdominal pain

Major Surgical or Invasive Procedure:
None

History of Present Illness:
Mr. ___ is a ___ male with PMHx significant for
stage IIIb supraclavicular melanoma s/p supraclavicular and
[...]
He also complains of left ankle pain after a fall and has been
taking ibuprofen.

Past Medical History:
1.right acoustic neuroma (deafness right ear)
2.s/p repair right biceps tendon rupture (___)
[...]
2. Bicep tendon repair.
3. Acoustic neuroma followed with serial MRIs.

Social History:
___
Family History:
FAMILY HISTORY:
1. Sister with uterine cancer.
[...]
5. Maternal aunt with colon cancer.
6. Maternal first cousin with melanoma.

Physical Exam:
Admission Physical Exam:
VS:
98.2 F 72 HR 140/67 20 98% RA
[...]
palpation.
Skin: No gross abnormalities noted

Discharge Physical Exam:
VS: 98.4, 75, 120/72, 24, 95 RA
GEN: Well appearing, sitting up in bed with family at bedside.
[...]
Neuro: A&Ox3. Follows commands and moves all extremities
equal
and strong. speech is clear and fluent.

Pertinent Results:
___ 10:30AM BLOOD WBC-12.3* RBC-3.81* Hgb-11.2* Hct-34.0*
MCV-89 MCH-29.4 MCHC-32.9 RDW-13.0 RDWSD-42.2 Plt ___
[...]
agreement with the discharge plan. He was instructed to follow
up with a colonoscopy outpatient in ___.

[Brief Hospital Course]

Medications on Admission:
tadalafil (CIALIS) 5 mg daily PRN
indomethacin 25 mg capsule TID

Discharge Medications:
1. Acetaminophen 1000 mg PO TID
Do not exceed 4 grams/ 24 hours
[...]
RX *metronidazole 500 mg 1 tablet(s) by mouth three times a day
Disp #*30 Tablet Refills:*0

Discharge Disposition:
Home

Discharge Diagnosis:
Perforated appendicitis

Discharge Condition:
Mental Status: Clear and coherent.
Level of Consciousness: Alert and interactive.
Activity Status: Ambulatory - Independent.

[Discharge Instructions]

Followup Instructions:
___

Figure 2: An example of the EHR with the location of the target discharge summaries. To show the sections used for the input text, the rounded rectangle is for the "Brief Hospital Course", the dashed rounded rectangle is for the "Discharge Instructions", and the rectangles are for both targets. The symbol "[...]" indicates omissions.

task.

Encoder-decoder. ClinicalT5 (Lu et al., 2022; Lehman et al., 2023), whose base model is T5 (Raffel et al., 2020), is the model pre-trained on clinical texts[1]. Lu et al. (2022) performed additional pre-training of the SciFive-PubMed-PMC (Phan et al., 2021) model on MIMIC-III (Johnson et al., 2016). Meanwhile, Lehman et al. (2023) pre-trained T5 from scratch using MIMIC-III and MIMIC-IV (Johnson et al., 2023c).

2.2 Clinical text summarization

Discharge Summarization. Williams et al. (2024) showed that although 33% of the discharge summaries generated by GPT-4 (Achiam et al., 2023) from the EHR were error-free, some contained hallucinations and omitted relevant information. Note, however, that the shared task does not allow data to be sent to third parties via an API.

Problem List Summarization (ProbSum). ProbSum (Gao et al., 2022) is a task aimed at generating a list of problems in a patient's daily care plan based on hospital records. In the BioNLP 2023 shared task (Gao et al., 2023) focused on ProbSum, the ensemble of ClinicalT5 models demonstrated robust performance (Manakul et al., 2023), and the approach combining Flan-T5 (Chung et al., 2022) with GPT2XL (Radford et al., 2019) also yielded strong results (Li et al., 2023). In the experiments using the shared task dataset, LLMs adapted to the medical domain demonstrated performance equal to or better than medical experts (Van Veen et al., 2024).

3 Task overview

3.1 Task description

Participants use an EHR dataset from MIMIC-IV (Johnson et al., 2023c) and develop a pipeline to generate two discharge summaries: the "Brief Hospital Course" section for patients and the "Discharge Instructions" section for clinicians. Table 1 shows an example of both sections.

3.2 Dataset description

The original datasets (Xu, 2024) include training, validation, phase I test, and phase II test sets. Participants use the training and validation sets to develop their pipeline, with the final evaluation per-

[1] Both of Lu et al. (2022) and Lehman et al. (2023) refer to their models as ClinicalT5.

Brief Hospital Course	Discharge Instructions
Mr. ___ is a ___ yo M with medical history significant for \\ stage IIIb supraclavicular melanoma and prostate cancer admitted \\ to the Acute Care Surgery Service on ___ with worsening \\ abdominal pain, frequent stools, and subjective fevers. He was \\ transferred from ___ for further management with a CT \\ abdomen showing a 5 x 6 x 7 cm right mid abdominal inflammatory \\ phlegmon. He was admitted to the surgical floor for IV \\ antibitoics and further evaluation.\\ \\ Gastroenterology was consulted for duodenal thickening. Given \\ his current infection the wall thickening is likely secondary to \\ the infection. Repeat imaging was recommended to evaluate \\ evolution of the phlegmon as well as outpatient colonoscopy once \\ antibiotic treatment is complete. \\ \\ The remainder of the hospital course is summarized below:\\ Neuro: The patient was alert and oriented throughout \\ hospitalization; pain was initially managed with a IV dilaudid. \\ He had left ankle pain and swelling consistent with gout that \\ was managed with PO indomethacin.. \\ CV: The patient remained stable from a cardiovascular \\ standpoint; vital signs were routinely monitored.\\ Pulmonary: The patient remained stable from a pulmonary \\ standpoint. Good pulmonary toilet, early ambulation and \\ incentive spirometry were encouraged throughout hospitalization. \\ \\ GI/GU/FEN: The patient was initially kept NPO. On HD3 he was \\ given a clear liquid diet. On HD4 he was advanced to regular \\ diet with good tolerability. Patient's intake and output were \\ closely monitored\\ ID: The patient's fever curves were closely watched for signs of \\ infection, of which there were none. He was initially given IV \\ zosyn and transitioned to oral flagyl and ciprofloxacin upon \\ discharge to complete a 2 week course of antibiotics. \\ HEME: The patient's blood counts were closely watched for signs \\ of bleeding, of which there were none.\\ Prophylaxis: The patient received subcutaneous heparin and ___ \\ dyne boots were used during this stay and was encouraged to get \\ up and ambulate as early as possible.\\ \\ At the time of discharge, the patient was doing well, afebrile \\ and hemodynamically stable. The patient was tolerating a diet, \\ ambulating, voiding without assistance, and pain was well \\ controlled. The patient received discharge teaching and \\ follow-up instructions with understanding verbalized and \\ agreement with the discharge plan. He was instructed to follow \\ up with a colonoscopy outpatient in ___.	Dr. ___,\\ \\ You were admitted to the Acute Care Surgery Service on ___ \\ with abdominal pain. You had a CT scan of your abdomen that \\ showed likely a perforated appendicitis. You were given IV \\ antibiotics and had improvement in your symptoms. An attempt was \\ made to drain the infection but it is not amenable to a drain at \\ this time. You were transitioned to oral antibiotics with \\ continued good effect.\\ \\ While in the hospital you had a flair up of gout in your left \\ ankle. You were given indomethacin with improvement in your \\ symptoms.\\ \\ You are now doing better, tolerating a regular diet, and ready \\ to be discharged to home to continue your recovery.\\ \\ Please note the following discharge instructions:\\ \\ Please call your doctor or nurse practitioner or return to the \\ Emergency Department for any of the following:\\ *You experience new chest pain, pressure, squeezing or \\ tightness.\\ *New or worsening cough, shortness of breath, or wheeze.\\ *If you are vomiting and cannot keep down fluids or your \\ medications.\\ *You are getting dehydrated due to continued vomiting, diarrhea, \\ or other reasons. Signs of dehydration include dry mouth, rapid \\ heartbeat, or feeling dizzy or faint when standing.\\ *You see blood or dark/black material when you vomit or have a \\ bowel movement.\\ *You experience burning when you urinate, have blood in your \\ urine, or experience a discharge.\\ *Your pain in not improving within ___ hours or is not gone \\ within 24 hours. Call or return immediately if your pain is \\ getting worse or changes location or moving to your chest or \\ back.\\ *You have shaking chills, or fever greater than 101.5 degrees \\ Fahrenheit or 38 degrees Celsius.\\ *Any change in your symptoms, or any new symptoms that concern \\ you.\\ \\ Please resume all regular home medications, unless specifically \\ advised not to take a particular medication. Also, please take \\ any new medications as prescribed.\\ \\ Please get plenty of rest, continue to ambulate several times \\ per day, and drink adequate amounts of fluids.

Table 1: An example of the "Brief Hospital Course" and "Discharge Instructions" sections. "\\" means line breaks.

Section	Prompt	Brief Hospital Course	Discharge Instructions
Name	The patient's name is provided as follows:	1	1
Sex	Gender details are as follows:	2	2
Service	The service details are as follows:	9	9
Allergies	Information on any allergies is detailed as follows:	7	6
Chief Complaint	The primary reason for the visit is summarized as follows:	3	3
Major Surgical or Invasive Procedure	Details on any major surgeries or invasive procedures are as follows:	8	7
History of Present Illness	An overview of the current illness's history is provided as follows:	4	4
Past Medical History	A summary of the patient's past medical history is as follows:	6	5
Pertinent Results	Clinically significant findings impacting the treatment and diagnosis are as follows:	5	–
Medications on Admission	Medications upon admission are detailed as follows:	–	8
Discharge Diagnosis	The final diagnosis at discharge is as follows:	–	10
Discharge Disposition	The disposition at discharge is provided as follows:	–	11
Discharge Condition	The patient's condition upon discharge is described as follows:	–	12
Discharge Medications	Medications prescribed at discharge are as follows:	–	13

Table 2: Prompts for each section and their priorities in each target discharge summary. The priority is used to order the sections in the input text.

formed on a subset of 250 samples from the phase II test set. See Appendix A for more details.

Note that although the datasets include metadata such as radiology reports in addition to the EHR and discharge summaries, we did not use this information in designing a simple pipeline. For more details, see the task website[2].

We created a new split with a 4:1 training-to-validation ratio using the original training and validation sets. Note that the EHR in the dataset contains the target texts: the "Brief Hospital Course" and the "Discharge Instructions" sections. As shown in Fig. 2, the "Brief Hospital Course" section is usually located in the middle of the discharge summary, while the "Discharge Instructions" section is generally located at the end of the EHR.

[2] https://stanford-aimi.github.io/discharge-me/

Rank	Team	Overall	BLEU	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	Meteor	AlignScore	MEDCON
1	WisPerMed	**0.332**	**0.124**	**0.453**	**0.201**	**0.308**	**0.438**	**0.403**	**0.315**	**0.411**
2	HarmonAI Lab at Yale	0.300	0.106*	0.423	0.180*	0.284	0.412	0.381	0.265*	0.353*
3	aehrc	0.297*	0.097	0.414	0.192	0.284	0.383*	0.398	0.274*	0.332*
4	EPFL-MAKE	0.289*	0.098	0.444	0.155	0.262*	0.399	0.336*	0.255*	0.360*
5	UF-HOBI	0.286*	0.102*	0.401*	0.174*	0.275*	0.395	0.289	0.296	0.355*
6	de ehren	0.284*	0.097	0.404*	0.166*	0.265*	0.389*	0.376	0.231	0.339*
7	DCT_PI	0.277*	0.092	0.401*	0.158	0.256*	0.378*	0.363	0.247	0.320
8	IgnitionInnovators	0.253	0.068	0.370*	0.131	0.245	0.360*	0.314	0.215	0.324
9	Shimo Lab (Ours)	0.248	0.063	0.394*	0.131	0.252*	0.351*	0.312	0.210	0.276
10	qub-cirdan	0.221	0.024	0.377*	0.106	0.205	0.300	0.332*	0.174	0.254
11	Roux-lette	0.206	0.030	0.319	0.084	0.182	0.289	0.287	0.195	0.265
12	UoG Siephers	0.191	0.017	0.341	0.109	0.209	0.268	0.247	0.143	0.193
13	mike-team	0.188	0.022	0.290	0.076	0.163	0.258	0.294	0.182	0.223
14	Ixa-UPV	0.183	0.016	0.259	0.057	0.144	0.282	0.284	0.210	0.215
15	MLBMIKABR	0.170	0.039	0.210	0.092	0.131	0.186	0.306	0.205	0.191
16	cyq	0.104	0.002	0.197	0.016	0.106	0.179	0.106	0.132	0.091
17	AIMI-Baseline	0.102	0.015	0.126	0.052	0.113	0.138	0.089	0.167	0.121

Table 3: The evaluation metrics values for the final test data. The higher values are better, and the highest value is highlighted in **bold**. Values that are at least 90% of the highest value are underlined, and values that are at least 80% of the highest value are marked with $(*)$.

Brief Hospital Course	Discharge Instructions
Mr. ___ is a __ yo M with PMHx significant for stage IIIb supraclavicular melanoma s/p supraclavicle and right anterior neck dissection and prostate cancer presenting with abdominal pain. The remainder of the hospital course is summarized below: - Neuro: The patient was alert and oriented throughout hospitalization; pain was initially managed with a IV dilaudid. He had left ankle pain and swelling consistent with gout that was managed with PO indomethacin. CV: The patient remained stable from a cardiovascular standpoint; vital signs were routinely monitored. Pulmonary: - The patient stayed stable from an pulmonary standpoint. Good pulmonary toilet, early ambulation and incentive spirometry were encouraged throughout hospitalization - GI/GU/FEN: The patient is initially kept NPO. On HD3 he was given a clear liquid diet. On HD4 he was advanced to regular diet with good tolerability. - Patient's intake and output were closely monitored ID: The patient's fever curves were closely watched for signs of infection, of which there were none. He was initially given IV zosyn and transitioned to oral flagyl and ciprofloxacin upon discharge to complete a 2 week course of antibiotics. HEME: The patient received subcutaneous heparin and dyne boots were used during this stay and was encouraged to get up and ambulate as early as possible. At the time of discharge, the patient was doing well, afebrile and hemodynamically stable. The patient was tolerating a diet, ambulating, voiding without assistance, and pain was well controlled. The patient received discharge teaching and follow-up instructions with understanding verbalized and agreement with the discharge plan. He was instructed to follow up with a colonoscopy outpatient in ________________________________	Dear Mr. ___, You were admitted to the hospital with abdominal pain. You were found to have a perforated appendicitis. You were treated with bowel rest and intravenous antibiotics. You are now ready to be discharged home to continue your recovery with the following instructions: Please call your doctor or nurse practitioner or return to the Emergency Department for any of the following: *You experience new chest pain, pressure, squeezing or tightness. *New or worsening cough, shortness of breath, or wheeze. *If you are vomiting and cannot keep down fluids or your medications. *You are getting dehydrated due to continued vomiting, diarrhea, or other reasons. Signs of dehydration include dry mouth, rapid heartbeat, or feeling dizzy or faint when standing. *You see blood or dark/black material when you vomit or have a bowel movement. *You experience burning when you urinate, have blood in your urine, or experience a discharge. *Your pain in not improving within 12 hours or is not gone within 24 hours. Call or return immediately if your pain is getting worse or changes location or moving to your chest or back. *You have shaking chills, or fever greater than 101.5 degrees Fahrenheit or 38 degrees Celsius. *Any change in your symptoms, or any new symptoms that concern you. Please resume all regular home medications, unless specifically advised not to take a particular medication. Also, please take any new medications as prescribed. Please get plenty of rest, continue to ambulate several times per day, and drink adequate amounts of fluids. Avoid lifting weights greater than __- lbs until you follow-up with your surgeon. Avoid driving or operating heavy machinery while taking pain medications.

Table 4: Our generated texts for the "Brief Hospital Course" and "Discharge Instructions" sections in Table 1.

3.3 Evaluation metrics

In this task, the following eight evaluation metrics[3] are used to compare the generated texts with the target texts: BLEU-4 (Papineni et al., 2002), ROUGE-1, ROUGE-2, ROUGE-L (Lin, 2004), BERTScore (Zhang et al., 2020), METEOR (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), MEDCON (Yim et al., 2023). The overall score is calculated by first averaging the scores for each target, and then averaging these values.

4 Pipeline

4.1 Input text preparation

We removed the target discharge summaries from the EHR as preprocessing. As shown in Fig. 2, the EHR contains redundant line breaks and detailed data. When the EHR is used directly as input text, this redundancy can increase the length of the input text. To mitigate this, we removed the noise from the EHR and selectively extracted the relevant sections for each target, thus avoiding the excessive length of the input text[4]. These sections were selected by excluding those with detailed data, such as timestamps[5], or those without specific information, such as the "Admission Date" section. Note that, in the case of preparing the input text for the model generating the "Brief Hospital Course" section, given the actual workflow of writing discharge summaries, we did not use the sections following this section in the input text.

For sections extracted from the EHR, we added an explanatory prompt to the beginning of each section and then concatenated the sections using the

[3] https://github.com/Stanford-AIMI/discharge-me/tree/main/scoring.

[4] The criteria for section selection are ad hoc, as mentioned in the Limitations section.

[5] Although the "Pertinent Results" section contains timestamps, we exclude them and use this section as input for the "Brief Hospital Course" section. See the Appendix B.3 for details.

`<sep>` tokens to create the final input text. Table 2 shows the prompts and priorities of the selected sections used in the input text for each target discharge summary. The sections in the input text were ordered according to the specified priorities, rather than their original order in the EHR. The input text was truncated if it exceeded the maximum text length[6].

In Appendix B, examples of input texts are shown in Tables 6 and 7, respectively, for "Brief Hospital Course" and "Discharge Instructions". These input texts were prepared from the EHR in Fig. 2. Histograms of the length of the input text are shown in Fig. 3.

4.2 Target text preparation

As shown in Table 1, the target texts contain many unnecessary line breaks. To prevent the line breaks from hindering the learning of the model, we removed them during preprocessing. In Appendix C, the texts before and after preprocessing for "Brief Hospital Course" are shown in Table 9, and those for "Discharge Instructions" are shown in Table 10. Histograms of the length of the target text are shown in Fig. 4.

4.3 Text generation

Using the input and target texts prepared in Sections 4.1 and 4.2, we performed LoRA (Hu et al., 2022) fine-tuning on the ClinicalT5-large[7] model published by Lu et al. (2022). The ClinicalT5-large model has 770M parameters with 24 layers. In Appendix D, the hyperparameters for fine-tuning and LoRA are shown in Tables 13 and 14. The hyperparameters to generate each target discharge summary are shown in Table 15.

5 Experiments

5.1 Results for the final test data

Table 3 presents the evaluation metrics values of the participating teams for the final test data. While our method did not achieve the highest scores of *WisPerMed* (Damm et al., 2024), it demonstrated relatively good performance in ROUGE-1, ROUGE-L, and BERTScore. In particular, we achieved a ROUGE-1 score of 0.394, which is comparable to top solutions such as those of *HarmonAI Lab at Yale* and *aehrc*.

5.2 Qualitative observation

Table 4 presents the summaries generated by our pipeline from the EHR for the target summaries in Table 1. While the detailed progress reports and discharge instructions may differ, the overall gist remains the same. In addition, unnecessary line breaks that were present in the original target summaries do not appear in the generated summaries.

6 Conclusion

We presented our approach to the shared task "Discharge Me!" at the BioNLP Workshop 2024. Extracting the relevant sections from the EHR, we added explanatory prompts to these sections and concatenated them with `<sep>` tokens to create the input text. We then performed LoRA fine-tuning on the ClinicalT5-large model. On the final test data, our approach achieved a ROUGE-1 score of 0.394, which is comparable to the top solutions.

Limitations

- Our pipeline cannot be applied to an EHR with different formats, resulting in a lack of generalizability. In fact, even in this shared task dataset, the lack of consistency in the original data sometimes makes it impossible to extract sections, resulting in incomplete summaries.
- When preparing the input text, adding prompts for each extracted section results in a longer length than simply concatenating sections with `<sep>` tokens.
- The effectiveness of our pipeline is not tested against other text generation models such as BioMistral-7B (Labrak et al., 2024) and the ClinicalT5-large model published by Lehman et al. (2023).
- While the selection and prioritization of the EHR sections used in the input text is somewhat ad-hoc, since extensive experiments would be required to compare the selection and prioritization, we did not conduct them in this study due to time and resource constraints.
- While the cleaned target texts are used for training, the original target texts with many line breaks are used for evaluation. This leads to a discrepancy between the target text distributions of training and evaluation.

[6] 1596 tokens

[7] `https://huggingface.co/luqh/ClinicalT5-large`

Ethics Statement

We conducted our research with careful consideration of data use and in accordance with the Data Use Agreement[8]. It is prohibited to identify individuals or organizations from the examples presented in the paper.

Acknowledgements

We would like to thank the Program Committee for their thorough review and valuable suggestions. This work was supported by JST SPRING, Grant Number JPMJSP2110. This study was partially supported by JST CREST JPMJCR21N3.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization@ACL 2005, Ann Arbor, Michigan, USA, June 29, 2005*, pages 65–72. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416.

Hendrik Damm, Tabea Margareta Grace Pakull, Bahadir Eryilmaz, Helmut Becker, Ahmad Idrissi-Yaghir, Henning Schäfer, Sergej Schultenkämper, and Christoph M. Friedrich. 2024. Wispermed at "discharge me!": Advancing text generation in healthcare with large language models, dynamic expert selection, and priming techniques on MIMIC-IV. *CoRR*, abs/2405.11255.

Hang Dong, Matús Falis, William Whiteley, Beatrice Alex, Joshua Matterson, Shaoxiong Ji, Jiaoyan Chen, and Honghan Wu. 2022. Automated clinical coding: what, why, and where we are? *npj Digit. Medicine*, 5.

Yanjun Gao, Dmitriy Dligach, Timothy Miller, and Majid Afshar. 2023. Overview of the problem list summarization (probsum) 2023 shared task on summarizing patients' active diagnoses and problems from electronic health record progress notes. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks, BioNLP@ACL 2023, Toronto, Canada, 13 July 2023*, pages 461–467. Association for Computational Linguistics.

Yanjun Gao, Dmitriy Dligach, Timothy Miller, Dongfang Xu, Matthew M. Churpek, and Majid Afshar. 2022. Summarizing patients' problems from hospital progress notes using pre-trained sequence-to-sequence models. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pages 2979–2991. International Committee on Computational Linguistics.

Ary L. Goldberger, Luis A. N. Amaral, Leon Glass, Jeffrey M. Hausdorff, Plamen Ch. Ivanov, Roger G. Mark, Joseph E. Mietus, George B. Moody, Chung-Kang Peng, and H. Eugene Stanley. 2000. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *Circulation*, 101(23):E215–E220.

Essam H. Houssein, Rehab E. Mohamed, and Abdelmgeid A. Ali. 2023. Heart disease risk factors detection from electronic health records using advanced nlp and deep learning techniques. *Scientific Reports*, 13(1):7173.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Leo Anthony Celi, Roger Mark, and Steven Horng. 2023a. MIMIC-IV-ED (version 2.2).

Alistair Johnson, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. 2023b. MIMIC-IV-Note: Deidentified free-text clinical notes (version 2.2).

Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, Li-wei H. Lehman, Leo A. Celi, and Roger G. Mark. 2023c. Mimic-iv, a freely accessible electronic health record dataset. *Sci Data*, 10(1):1. Erratum in: Sci

[8] `https://physionet.org/content/discharge-me/view-dua/1.3/`

Data. 2023 Jan 16;10(1):31. Erratum in: Sci Data. 2023 Apr 18;10(1):219.

Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Liwei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. 2016. Mimic-iii, a freely accessible critical care database. *Sci Data*, 3:160035.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. Biomistral: A collection of open-source pretrained large language models for medical domains. *CoRR*, abs/2402.10373.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model.

Eric Lehman, Evan Hernandez, Diwakar Mahajan, Jonas Wulff, Micah J. Smith, Zachary Ziegler, Daniel Nadler, Peter Szolovits, Alistair E. W. Johnson, and Emily Alsentzer. 2023. Do we still need clinical language models? In *Conference on Health, Inference, and Learning, CHIL 2023, Broad Institute of MIT and Harvard (Merkin Building), 415 Main Street, Cambridge, MA, USA*, volume 209 of *Proceedings of Machine Learning Research*, pages 578–597. PMLR.

Hao Li, Yu-Ping Wu, Viktor Schlegel, Riza Batista-Navarro, Thanh-Tung Nguyen, Abhinav Ramesh Kashyap, Xiao-Jun Zeng, Daniel Beck, Stefan Winkler, and Goran Nenadic. 2023. Team: PULSAR at probsum 2023: PULSAR: pre-training with extracted healthcare terms for summarising patients' problems and data augmentation with black-box large language models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks, BioNLP@ACL 2023, Toronto, Canada, 13 July 2023*, pages 503–509. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Jinghui Liu, Daniel Capurro, Anthony Nguyen, and Karin Verspoor. 2022. "note bloat" impacts deep learning-based nlp models for clinical prediction tasks. *Journal of biomedical informatics*, 133:104149.

Qiuhao Lu, Dejing Dou, and Thien Huu Nguyen. 2022. Clinicalt5: A generative language model for clinical text. In *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 5436–5443. Association for Computational Linguistics.

Potsawee Manakul, Yassir Fathullah, Adian Liusie, Vyas Raina, Vatsal Raina, and Mark J. F. Gales. 2023. CUED at probsum 2023: Hierarchical ensemble of summarization models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks, BioNLP@ACL 2023, Toronto, Canada, 13 July 2023*, pages 516–523. Association for Computational Linguistics.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`.

Nir Menachemi and Taleah H Collum. 2011. Benefits and drawbacks of electronic health record systems. *Risk Manag Healthc Policy*, 4:47–55.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Long N. Phan, James T. Anibal, Hieu Tran, Shaurya Chanana, Erol Bahadroglu, Alec Peltekian, and Grégoire Altan-Bonnet. 2021. Scifive: a text-to-text transformer model for biomedical literature. *CoRR*, abs/2106.03598.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Richard J Roberts. 2001. Pubmed central: The genbank of the published literature.

Tait D Shanafelt, Lotte N Dyrbye, Christine Sinsky, Omar Hasan, Daniel Satele, Jeff Sloan, and Colin P West. 2016. Relationship between clerical burden and characteristics of the electronic environment with physician burnout and professional satisfaction. *Mayo Clin Proc*, 91(7):836–848.

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, Eduardo Pontes Reis, Anna Seehofnerová, et al. 2024. Adapted large language models can outperform medical experts in clinical text summarization. *Nature Medicine*, pages 1–9.

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, Eduardo Pontes Reis, Anna Seehofnerova, Nidhi Rohatgi, Poonam Hosamani, William Collins, Neera Ahuja, Curtis P. Langlotz, Jason Hom, Sergios Gatidis, John Pauly, and Akshay S. Chaudhari. 2023. Clinical text summarization: Adapting large language models can outperform human experts. *Res Sq*. Preprint: rs.3.rs-3483777.

Guangyu Wang, Guoxing Yang, Zongxin Du, Longjun Fan, and Xiaohu Li. 2023. Clinicalgpt: Large language models finetuned with diverse medical data and comprehensive evaluation. *CoRR*, abs/2306.09968.

Christopher YK Williams, Jaskaran Bains, Tianyu Tang, Kishan Patel, Alexa N Lucas, Fiona Chen, Brenda Y Miao, Atul J Butte, and Aaron E Kornblith. 2024. Evaluating large language models for drafting emergency department discharge summaries. *medRxiv*, pages 2024–04.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos, Online, November 16-20, 2020*, pages 38–45. Association for Computational Linguistics.

Justin Xu. 2024. Discharge me: Bionlp acl'24 shared task on streamlining discharge documentation. `https://doi.org/10.13026/4a0k-4360`. Version 1.2.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10(1):586.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with A unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 11328–11348. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Dataset	Numbers
Training	68,785
Validation	14,719
Phase I Test	14,702
Phase II Test	10,962
Total	109,168

Table 5: The number of samples for the data splits.

A Details of datasets

In this task, we use the dataset created by the MIMIC-IV's submodules MIMIC-IV-ED (Johnson et al., 2023a) and MIMIC-IV-Note (Johnson et al., 2023b). The dataset is available on PhysioNet (Goldberger et al., 2000), and its use requires completion of the CITI[9] training and credentialing process. Table 5 lists the number of samples for the data splits.

B Details of input text

This section first explains the detailed preprocessing required to create input text from the EHR. It

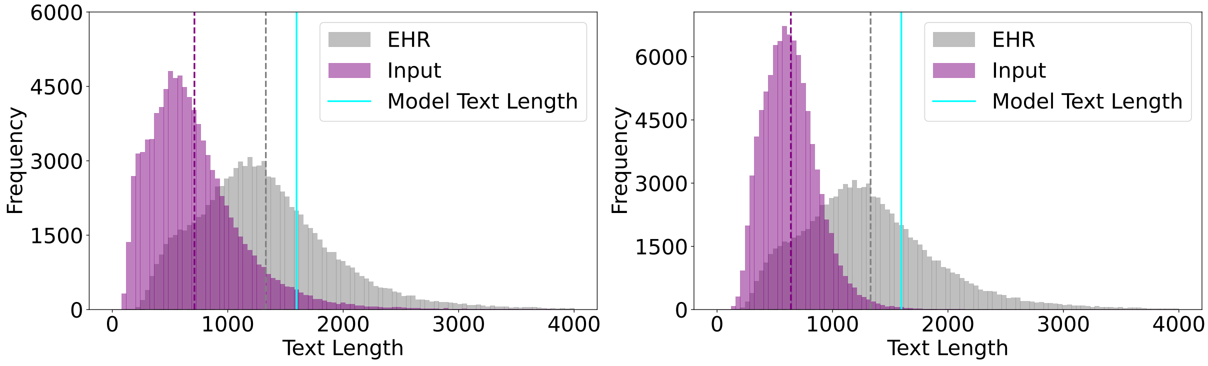
[9] `https://about.citiprogram.org/`

The patient's name is provided as follows: ___.<sep>Gender details are as follows: Male.<sep>The primary reason for the visit is summarized as follows: Abdominal pain.<sep>An overview of the current illness's history is provided as follows: Mr. ___ is a ___ male with PMHx significant for stage IIIb supraclavicular melanoma s/p supraclavicular and right anterior neck dissection and prostate cancer s/p radical prostatectomy, presenting with abdominal pain. The pain began two weeks ago and has been worsening in the last few days. His pain is localized to the right periumbilical region. He endorses having chills, inability to sleep and eat due to pain and a 6 lb weight loss in the past few days. He has been passing flatus and having loose and frequent non-bloody stools. He has also been having night sweats. He denies fever, nausea or vomiting. He was seen at ___ and transferred to ___ ED for further management after his CT abdomen showed a 5 x 6 x 7 cm right mid abdominal inflammatory phlegmon. His last colonoscopy was ___ years ago which revealed some polyps. He also complains of left ankle pain after a fall and has been taking ibuprofen.<sep>Clinically significant findings impacting the treatment and diagnosis are as follows: MCV-89 MCH-29.4 MCHC-32.9 RDW-13.0 RDWSD-42.2 Plt ___ MCV-88 MCH-29.8 MCHC-34.1 RDW-12.7 RDWSD-40.3 Plt ___ MCV-88 MCH-29.9 MCHC-33.8 RDW-12.5 RDWSD-40.2 Plt ___ MCV-88 MCH-29.5 MCHC-33.5 RDW-12.4 RDWSD-39.8 Plt ___ K-3.8 Cl-103 HCO3-24 AnGap-18 K-3.6 Cl-99 HCO3-24 AnGap-20 K-3.5 Cl-97 HCO3-24 AnGap-19 K-3.8 Cl-96 HCO3-21* AnGap-22* Glucose-NEG Ketone-80 Bilirub-NEG Urobiln-NEG pH-6.0 Leuks-NEG Epi-0 **FINAL REPORT ___. URINE CULTURE (Final ___: NO GROWTH. RADIOLOGY: * Phlegmon/ multiloculated fluid collection with surrounding. extensive inflammatory changes is indistinguishable from the distal portion of the appendix. Findings are concerning for perforated appendicitis. Possibility of underlying mass is difficult to exclude, particularly in this patient with history of melanoma. * Duodenal wall thickening may be inflammatory secondary to the. adjacent phlegmon. * Duodenum does not cross the midline, consistent with. intestinal malrotation. * Cholelithiasis. * Nonspecific bulbous appearance of the uncinate process of the. pancreas without discrete lesion identified. No pancreatic ductal dilatation. * Unorganized fluid/phlegmonous collection within the right. lower quadrant, surrounding the appendix, appears minimally enlarged since the reference study from ___. The findings favor ruptured appendicitis or a ruptured appendiceal mucocele. A neoplastic source relating to known history of melanoma would be atypical. Continued short-term imaging surveillance is recommended. * Congenital bowel malrotation, without volvulus or. obstruction. * Cholelithiasis. The remainder of the hospital course is summarized below: Neuro: The patient was alert and oriented throughout hospitalization; pain was initially managed with a IV dilaudid. He had left ankle pain and swelling consistent with gout that was managed with PO indomethacin.. CV: The patient remained stable from a cardiovascular standpoint; vital signs were routinely monitored. Pulmonary: The patient remained stable from a pulmonary standpoint. Good pulmonary toilet, early ambulation and incentive spirometry were encouraged throughout hospitalization. GI/GU/FEN: The patient was initially kept NPO. On HD3 he was given a clear liquid diet. On HD4 he was advanced to regular diet with good tolerability. Patient's intake and output were closely monitored. ID: The patient's fever curves were closely watched for signs of infection, of which there were none. He was initially given IV zosyn and transitioned to oral flagyl and ciprofloxacin upon discharge to complete a 2 week course of antibiotics. HEME: The patient's blood counts were closely watched for signs of bleeding, of which there were none. Prophylaxis: The patient received subcutaneous heparin and ___ dyne boots were used during this stay and was encouraged to get up and ambulate as early as possible. At the time of discharge, the patient was doing well, afebrile and hemodynamically stable. The patient was tolerating a diet, ambulating, voiding without assistance, and pain was well controlled. The patient received discharge teaching and follow-up instructions with understanding verbalized and agreement with the discharge plan. He was instructed to follow up with a colonoscopy outpatient in ___.<sep>A summary of the patient's past medical history is as follows: 1.right acoustic neuroma (deafness right ear) 2.s/p repair right biceps tendon rupture (___) 3.s/p right supraclavicular lymph node biopsy (___). PAST MEDICAL HISTORY: Stage IIIb melanoma diagnosed in ___ with findings of a positive right supraclavicular node, status post right anterior neck dissection revealing ___ additional positive nodes. He had adjuvant interferon therapy with Dr. ___ completed in ___, 36 weeks of this treatment. Bicep tendon repair. Acoustic neuroma followed with serial MRIs.<sep>Information on any allergies is detailed as follows: Codeine / Levaquin.<sep>Details on any major surgeries or invasive procedures are as follows: None.<sep>The service details are provided as follows: SURGERY.

Table 6: Input text from the EHR shown in Fig. 2 to generate the "Brief Hospital Course" section. The prompts used in both targets are highlighted in green and the prompt used only for "Brief Hospital Course" is highlighted in blue.

The patient's name is provided as follows: ___.<sep>Gender details are as follows: Male.<sep>The primary reason for the visit is summarized as follows: Abdominal pain.<sep>An overview of the current illness's history is provided as follows: Mr. ___ is a ___ male with PMHx significant for stage IIIb supraclavicular melanoma s/p supraclavicular and right anterior neck dissection and prostate cancer s/p radical prostatectomy, presenting with abdominal pain. The pain began two weeks ago and has been worsening in the last few days. His pain is localized to the right periumbilical region. He endorses having chills, inability to sleep and eat due to pain and a 6 lb weight loss in the past few days. He has been passing flatus and having loose and frequent non-bloody stools. He has also been having night sweats. He denies fever, nausea or vomiting. He was seen at ___ and transferred to ___ ED for further management after his CT abdomen showed a 5 x 6 x 7 cm right mid abdominal inflammatory phlegmon. His last colonoscopy was ___ years ago which revealed some polyps. He also complains of left ankle pain after a fall and has been taking ibuprofen.<sep>A summary of the patient's past medical history is as follows: 1.right acoustic neuroma (deafness right ear) 2.s/p repair right biceps tendon rupture (___) 3.s/p right supraclavicular lymph node biopsy (___). PAST MEDICAL HISTORY: Stage IIIb melanoma diagnosed in ___ with findings of a positive right supraclavicular node, status post right anterior neck dissection revealing ___ additional positive nodes. He had adjuvant interferon therapy with Dr. ___ completed in ___, 36 weeks of this treatment. Bicep tendon repair. Acoustic neuroma followed with serial MRIs.<sep>Information on any allergies is detailed as follows: Codeine / Levaquin.<sep>Details on any major surgeries or invasive procedures are as follows: None.<sep>Medications upon admission are detailed as follows: tadalafil (CIALIS) 5 mg daily PRN indomethacin 25 mg capsule TID.<sep>The service details are provided as follows: SURGERY.<sep>The final diagnosis at discharge is as follows: Perforated appendicitis.<sep>The disposition at discharge is provided as follows: Home.<sep>The patient's condition upon discharge is described as follows: Mental Status is Clear and coherent. Level of Consciousness is Alert and interactive. Activity Status is Ambulatory - Independent.<sep>Medications prescribed at discharge are as follows: * Acetaminophen 1000 mg PO TID. Do not exceed 4 grams/ 24 hours. * Ciprofloxacin HCl 500 mg PO Q12H. monitor for s/sx of allergic reaction RX *ciprofloxacin HCl 500 mg 1 tablet(s) by mouth twice a day. Disp #*20 Tablet Refills:*0 * Indomethacin 25 mg PO TID. RX *indomethacin 25 mg 1 capsule(s) by mouth three times a day. Disp #*42 Capsule Refills:*0 * MetroNIDAZOLE 500 mg PO Q8H. RX *metronidazole 500 mg 1 tablet(s) by mouth three times a day. Disp #*30 Tablet Refills:*0.

Table 7: Input text from the EHR in Fig. 2 to generate the "Discharge Instructions" section. The prompts used in both targets are highlighted in green and the prompts used only for "Discharge Instructions" are highlighted in orange.



(a) Brief Hospital Course

(b) Discharge Instructions

Figure 3: Histograms of the text length (in tokens) of the EHR and the input texts for the training and validation sets. The dashed line is the mean. The maximum text length is 1596 tokens, and see Table 12 in Appendix D for more details.

	Brief Hospital Course		Discharge instructions	
	EHR	Input	EHR	Input
Min	101	80	101	115
Max	8725	5664	8725	5774
Mean	1330	554	1330	639

Table 8: Statistical information (in tokens) for histograms in Fig. 3.

then provides examples and statistical information before and after preprocessing.

B.1 Extraction of simple sections

This section explains the process for extracting the “Sex”, “Service”, “Allergies”, “Chief Complaint”, and “Major Surgical or Invasive Procedure” sections.

To extract these sections, we used specific regular expressions such as `Sex: (\w+)\n`.

B.2 Extraction of complex sections

This section explains the process for extracting the “History of Present Illness”, “Past Medical History”, “Pertinent Results”, “Medications on Admission”, “Discharge Medications”, “Discharge Disposition”, “Discharge Diagnosis”, and ”Discharge Condition” sections.

We performed more detailed processing and pattern matching to efficiently extract the text of these sections. For example, for the “Discharge Condition” section, we used the regular expression `Discharge Diagnosis:\s*\n(.*?)(?=Discharge Condition:)` and it matches the diagnosis text up to the “Discharge Condition” section.

B.3 Detailed processing of each section

“Name”. The patient’s name is given as “___” and we used it directly.

“Sex”. We converted “M” to “Male” and “F” to “Female”.

“Pertinent Results”. Timestamps in lines like “__ 08:00AM BLOOD __” were removed using regular expressions. In addition, list sections are converted to “*” format to maintain text consistency and clarity.

“Medications on Admission”. List sections are converted to “*” format to maintain text consistency and clarity.

“Discharge Condition”. We changed a colon in the extracted text to “is”. For example, “Condition: Stable” is changed to “Condition is Stable”.

“Discharge Medications”. List sections are converted to “*” format to maintain text consistency and clarity.

B.4 Other processing

We ensure textual continuity by replacing line breaks with spaces and trimming excess spaces. In cases where no matching text is found, the default response is designated as “Unknown”.

B.5 Examples of input text

Tables 6 and 7 show examples of input text. These examples illustrate that the ClinicalT5-large model is fine-tuned with different input text for each target discharge summary.

B.6 Statistical information

Fig. 3 shows histograms of the text length (in tokens) of the EHR and the input texts for the training and validation sets. Table 8 shows the statistical information for these histograms. As shown in Fig. 3 and Table 8, the preprocessing significantly reduces the length of the text.

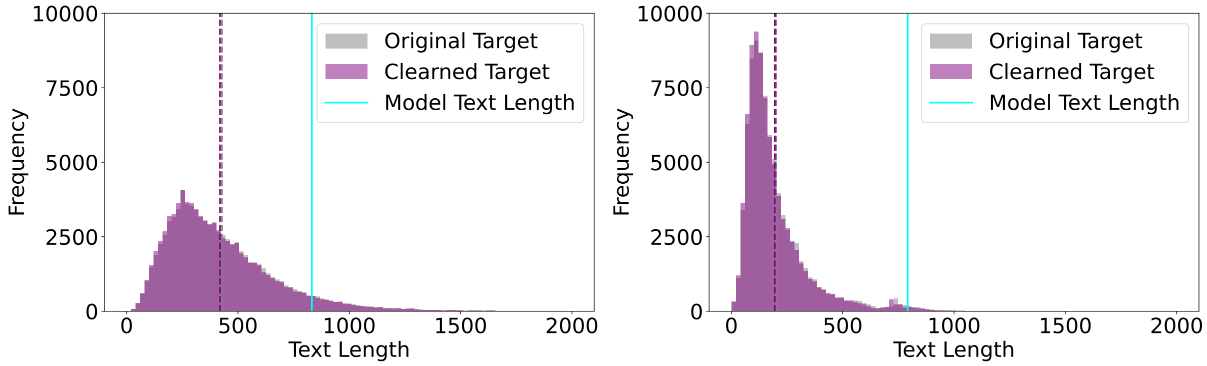
C Details of target text

C.1 Extraction and concatenation of segments

In the first process of segment extraction, we divide the text into segments based on blank lines and identify the distinct segments. We then remove spaces and line breaks from each segment and discard empty segments to retain only meaningful segments. Multiple consecutive spaces within each segment are replaced by a single space to improve readability. Finally, we reassemble the cleaned segments with line breaks to make them more suitable for training language models.

Text	Cleaned Text
Mr. ___ is a ___ yo M with medical history significant for \\ stage IIIb supraclavicular melanoma and prostate cancer admitted \\ to the Acute Care Surgery Service on ___ with worsening \\ abdominal pain, frequent stools, and subjective fevers. He was \\ transferred from ___ for further management with a CT \\ abdomen showing a 5 x 6 x 7 cm right mid abdominal inflammatory \\ phlegmon. He was admitted to the surgical floor for IV \\ antibitoics and further evaluation.\\ \\ Gastroenterology was consulted for duodenal thickening. Given \\ his current infection the wall thickening is likely secondary to \\ the infection. Repeat imaging was recommended to evaluate \\ evolution of the phlegmon as well as outpatient colonoscopy once \\ antibiotic treatment is complete. \\ \\ The remainder of the hospital course is summarized below:\\ Neuro: The patient was alert and oriented throughout \\ hospitalization; pain was initially managed with a IV dilaudid. \\ He had left ankle pain and swelling consistent with gout that \\ was managed with PO indomethacin.. \\ CV: The patient remained stable from a cardiovascular \\ standpoint; vital signs were routinely monitored.\\ Pulmonary: The patient remained stable from a pulmonary \\ standpoint. Good pulmonary toilet, early ambulation and \\ incentive spirometry were encouraged throughout hospitalization. \\ \\ GI/GU/FEN: The patient was initially kept NPO. On HD3 he was \\ given a clear liquid diet. On HD4 he was advanced to regular \\ diet with good tolerability. Patient's intake and output were \\ closely monitored\\ ID: The patient's fever curves were closely watched for signs of \\ infection, of which there were none. He was initially given IV \\ zosyn and transitioned to oral flagyl and ciprofloxacin upon \\ discharge to complete a 2 week course of antibiotics. \\ HEME: The patient's blood counts were closely watched for signs \\ of bleeding, of which there were none.\\ Prophylaxis: The patient received subcutaneous heparin and ___ \\ dyne boots were used during this stay and was encouraged to get \\ up and ambulate as early as possible.\\ \\ At the time of discharge, the patient was doing well, afebrile \\ and hemodynamically stable. The patient was tolerating a diet, \\ ambulating, voiding without assistance, and pain was well \\ controlled. The patient received discharge teaching and \\ follow-up instructions with understanding verbalized and \\ agreement with the discharge plan. He was instructed to follow \\ up with a colonoscopy outpatient in ___.	Mr. ___ is a ___ yo M with medical history significant for stage IIIb supraclavicular melanoma and prostate cancer admitted to the Acute Care Surgery Service on ___ with worsening abdominal pain, frequent stools, and subjective fevers. He was transferred from ___ for further management with a CT abdomen showing a 5 x 6 x 7 cm right mid abdominal inflammatory phlegmon. He was admitted to the surgical floor for IV antibitoics and further evaluation.\\ \\ Gastroenterology was consulted for duodenal thickening. Given his current infection the wall thickening is likely secondary to the infection. Repeat imaging was recommended to evaluate evolution of the phlegmon as well as outpatient colonoscopy once antibiotic treatment is complete.\\ \\ The remainder of the hospital course is summarized below:\\ Neuro: The patient was alert and oriented throughout hospitalization; pain was initially managed with a IV dilaudid. He had left ankle pain and swelling consistent with gout that was managed with PO indomethacin.. CV: The patient remained stable from a cardiovascular standpoint; vital signs were routinely monitored.\\ Pulmonary: The patient remained stable from a pulmonary standpoint. Good pulmonary toilet, early ambulation and incentive spirometry were encouraged throughout hospitalization.\\ \\ GI/GU/FEN: The patient was initially kept NPO. On HD3 he was given a clear liquid diet. On HD4 he was advanced to regular diet with good tolerability. Patient's intake and output were closely monitored\\ ID: The patient's fever curves were closely watched for signs of infection, of which there were none. He was initially given IV zosyn and transitioned to oral flagyl and ciprofloxacin upon discharge to complete a 2 week course of antibiotics. HEME: The patient's blood counts were closely watched for signs of bleeding, of which there were none.\\ Prophylaxis: The patient received subcutaneous heparin and ___ dyne boots were used during this stay and was encouraged to get up and ambulate as early as possible.\\ \\ At the time of discharge, the patient was doing well, afebrile and hemodynamically stable. The patient was tolerating a diet, ambulating, voiding without assistance, and pain was well controlled. The patient received discharge teaching and follow-up instructions with understanding verbalized and agreement with the discharge plan. He was instructed to follow up with a colonoscopy outpatient in ___.

Table 9: The text of the "Brief Hospital Course" section in Table 1 and its cleaned text by preprocessing. "\\" means line breaks.



(a) Brief Hospital Course

(b) Discharge Instructions

Figure 4: Histograms of the text length (in tokens) of the target texts before and after preprocessing for the training and validation sets. The dashed line is the mean. The maximum text length is 832 tokens for "Brief Hospital Course" and 792 tokens for "Discharge Instructions", see Table 12 in Appendix D for more details.

C.2 Examples of preprocessed target text

Tables 9 and 10 show examples of the target text before and after preprocessing. These examples illustrate that redundant line breaks are removed after preprocessing.

C.3 Statistical information

Fig. 4 shows histograms of the text length (in tokens) of the target texts before and after preprocessing for the training and validation sets. Table 11 shows the statistical information for these

Text	Cleaned Text
Dr. ___,\\ \\ You were admitted to the Acute Care Surgery Service on ___ \\ with abdominal pain. You had a CT scan of your abdomen that \\ showed likely a perforated appendicitis. You were given IV \\ antibiotics and had improvement in your symptoms. An attempt was \\ made to drain the infection but it is not amenable to a drain at \\ this time. You were transitioned to oral antibiotics with \\ continued good effect.\\ \\ While in the hospital you had a flair up of gout in your left \\ ankle. You were given indomethacin with improvement in your \\ symptoms.\\ \\ You are now doing better, tolerating a regular diet, and ready \\ to be discharged to home to continue your recovery.\\ \\ Please note the following discharge instructions:\\ \\ Please call your doctor or nurse practitioner or return to the \\ Emergency Department for any of the following:\\ *You experience new chest pain, pressure, squeezing or \\ tightness.\\ *New or worsening cough, shortness of breath, or wheeze.\\ *If you are vomiting and cannot keep down fluids or your \\ medications.\\ *You are getting dehydrated due to continued vomiting, diarrhea, \\ or other reasons. Signs of dehydration include dry mouth, rapid \\ heartbeat, or feeling dizzy or faint when standing.\\ *You see blood or dark/black material when you vomit or have a \\ bowel movement.\\ *You experience burning when you urinate, have blood in your \\ urine, or experience a discharge.\\ *Your pain in not improving within ___ hours or is not gone \\ within 24 hours. Call or return immediately if your pain is \\ getting worse or changes location or moving to your chest or \\ back.\\ *You have shaking chills, or fever greater than 101.5 degrees \\ Fahrenheit or 38 degrees Celsius.\\ *Any change in your symptoms, or any new symptoms that concern \\ you.\\ \\ Please resume all regular home medications, unless specifically \\ advised not to take a particular medication. Also, please take \\ any new medications as prescribed.\\ \\ Please get plenty of rest, continue to ambulate several times \\ per day, and drink adequate amounts of fluids.	Dr. ___,\\ \\ You were admitted to the Acute Care Surgery Service on ___ with abdominal pain. You had a CT scan of your abdomen that showed likely a perforated appendicitis. You were given IV antibiotics and had improvement in your symptoms. An attempt was made to drain the infection but it is not amenable to a drain at this time. You were transitioned to oral antibiotics with continued good effect.\\ \\ While in the hospital you had a flair up of gout in your left ankle. You were given indomethacin with improvement in your symptoms.\\ \\ You are now doing better, tolerating a regular diet, and ready to be discharged to home to continue your recovery.\\ \\ Please note the following discharge instructions:\\ \\ Please call your doctor or nurse practitioner or return to the Emergency Department for any of the following:\\ *You experience new chest pain, pressure, squeezing or tightness.\\ *New or worsening cough, shortness of breath, or wheeze.\\ *If you are vomiting and cannot keep down fluids or your medications.\\ *You are getting dehydrated due to continued vomiting, diarrhea, or other reasons. Signs of dehydration include dry mouth, rapid heartbeat, or feeling dizzy or faint when standing.\\ *You see blood or dark/black material when you vomit or have a bowel movement.\\ *You experience burning when you urinate, have blood in your urine, or experience a discharge.\\ *Your pain in not improving within ___ hours or is not gone within 24 hours. Call or return immediately if your pain is getting worse or changes location or moving to your chest or back.\\ *You have shaking chills, or fever greater than 101.5 degrees Fahrenheit or 38 degrees Celsius.\\ *Any change in your symptoms, or any new symptoms that concern you.\\ \\ Please resume all regular home medications, unless specifically advised not to take a particular medication. Also, please take any new medications as prescribed.\\ \\ Please get plenty of rest, continue to ambulate several times per day, and drink adequate amounts of fluids.

Table 10: The text of the "Discharge Instructions" section in Table 1 and its cleaned text by preprocessing. "\\" means line breaks.

	Brief Hospital Course		Discharge instructions	
	Original Target	Cleaned Target	Original Target	Cleaned Target
Min	2	1	10	10
Max	4614	4452	5025	4861
Mean	428	419	201	195

Table 11: Statistical information (in tokens) for histograms in Fig. 4.

	Brief Hospital Course	Discharge instructions
Input Text	1596	
Generated Text	832	792
Text	832	792

Table 12: Maximum text length (tokens).

histograms. As shown in Fig. 4 and Table 11, the preprocessing slightly reduces the length of the text.

D Details of fine-tuning

We used Pytorch (Paszke et al., 2019) and hugging-face transformers (Wolf et al., 2020) to implement and fine-tune our models. We also use peft (Man-

Batch size	2
Epochs	4
Learning rate	1e-4
Precision setting	FP16
Weight decay	0.01

Table 13: Hyperparameters for fine-tuning.

Dropout probability	0.05
Rank	4
Target modules	Query & Value
α	16

Table 14: Hyperparameters for LoRA.

Min length	10
Num beams	4
Do sample	True
Length penalty	1.1
No repeat n-gram size	4

Table 15: Hyperparameters to generate each target discharge summary.

grulkar et al., 2022) for LoRA.

Table 12 shows the text length (in tokens) used by our models. Table 13 shows the hyperparameters used for fine tuning. Table 14 shows the hyperparameters used for LoRA. Table 15 shows the hyperparameters to generate each target discharge summary.

Ixa-Med at Discharge Me! Retrieval-Assisted Generation for Streamlining Discharge Documentation

Jordan Koontz
HiTZ Center - Ixa
UPV/EHU
jkoontz001@ikasle.ehu.eus

Maite Oronoz
HiTZ Center - Ixa
UPV/EHU
maite.oronoz@ehu.eus

Alicia Pérez
HiTZ Center - Ixa
UPV/EHU
alicia.perez@ehu.eus

Abstract

In this paper we present our system for the BioNLP ACL'24 "Discharge Me!" task on automating discharge summary section generation. Using Retrieval-Augmented Generation, we combine a Large Language Model (LLM) with external knowledge to guide the generation of the target sections. Our approach generates structured patient summaries from discharge notes using an instructed LLM, retrieves relevant "Brief Hospital Course" and "Discharge Instructions" examples via BM25 and SentenceBERT, and provides this context to a frozen LLM for generation. Our top system using SentenceBERT retrieval achieves an overall score of 0.183, outperforming zero-shot baselines. We analyze performance across different aspects, discussing limitations and future research directions.

1 Introduction

Generating detailed clinical notes in Electronic Health Records (EHRs) is a time-consuming task that can lead to clinician burnout and operational inefficiencies in healthcare systems. The BioNLP ACL'24 Shared Task, "Discharge Me!" (Xu et al., 2024), aims to automate the generation of critical discharge summary sections using natural language processing (NLP). While large language models (LLMs) like GPT-4 (OpenAI et al., 2024) and Llama-3 (Meta, 2024) have advanced NLP capabilities, they can produce hallucinations when encountering out-of-distribution queries (Zhang et al., 2023).

The Retrieval-Augmented Generation (RAG) framework aims to mitigate hallucinations in large language models (LLMs) by combining external knowledge retrieval with LLM generation (Lewis et al., 2020; Ma et al., 2023). A Naive RAG approach involves indexing data into vectors, retrieving relevant vectors for a given query, and providing the retrieved context to a frozen LLM. However, this naive implementation often suffers from limitations in retrieval precision, recall, and generation quality. Notwithstanding, we evaluate the efficacy of a Naive RAG framework for the "Discharge Me!" task. Section 3 describes our system methodology and presents our results. Section 4 analyzes the limits of our approach and outlines prospective research areas for improvement.

2 Task Description

The BioNLP ACL'24 Shared Task, 'Discharge Me!", focuses on streamlining the clinical documentation process by automating the generation of two critical sections in discharge summaries: "Brief Hospital Course" and "Discharge Instructions". By reducing the time and effort clinicians expend on writing these detailed notes in electronic health records (EHRs), we can alleviate administrative burden, minimize clinician burnout, and ultimately improve operational efficiencies and patient care quality.

2.1 Dataset Description

For this shared task, participants are provided with a dataset derived from the MIMIC-IV-Note and MIMIC-IV-ED submodules. The shared task dataset contains $109,168$ visits to the Emergency Department (ED). Each visit consists of chief complaints documented by ED physicians, ICD diagnosis codes (either ICD-9 or ICD-10), at least one associated radiology report, and the full discharge summary text which includes the "Brief Hospital Stay" and "Discharge Instructions" sections, among others. The dataset is split into training ($68,785$ samples), validation ($14,719$ samples), phase I testing ($14,702$ samples), and phase II testing ($10,962$ samples). The chief goal is to develop a system that can generate the two target sections given the available data for each visit.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 658–663
August 16, 2024. ©2024 Association for Computational Linguistics

2.2 Evaluation

For evaluating the participants' systems, a hidden subset of 250 samples from the test phase I and test phase II is used. The evaluation framework is composed of a diverse array or metrics that capture both textual similarity and factual correctness aspects of the generated texts. Concretely, the following metrics are used: BLEU-4 (Papineni et al., 2002), ROUGE-1, ROUGE-2, ROUGE-L (Lin, 2004), BERTScore (Zhang et al., 2020), Meteor (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023). The final overall system score is a composite measure derived by combining the scores across all evaluation metrics and both target sections.

3 Methods & Results

3.1 Structured Patient Summary Generation

The first step in our approach involved generating structured JSON summaries from the patient discharge summaries. This process extracted and organized relevant information for generating the "Brief Hospital Course" and "Discharge Instructions" sections, critical components of discharge documentation..

We leveraged the capabilities of an LLM, specifically[1] the `mistralai/Mistral-7B-Instruct-v0.2` model, to facilitate this preprocessing step. The `vllm` (Kwon et al., 2023) library was utilized for interacting with the LLM, while the `lmformatenforcer`[2] library ensured character-level parsing and schema enforcement.

Our pipeline consisted of the following steps:

1. **Data Masking**: To ensure that the LLM generated summaries based solely on the available information, we masked the "Discharge Instructions" and "Brief Hospital Course" sections from the input discharge summaries.

2. **Prompt and Schema Design**: A carefully crafted prompt template, presented in table 1, was designed to guide the LLM in generating structured JSON summaries. Additionally, we defined a Pydantic data model to serve as the schema for the desired JSON output format.

3. **LLM Inference**: For each masked discharge summary, we employed the LLM to generate two structured JSON summaries using the defined prompt template. One summary excluded the "Discharge Instructions section", while the other omitted the "Brief Hospital Course section".

The structured summaries (mentioned in step 2) aimed to captured essential patient information like demographics, medical history, reason for admission, findings, treatments, and discharge condition. This structured input aimed to reduce noise and prevent hallucinations in subsequent generation steps.

3.2 Zero-shot Generation

We first established a baseline by conducting experiments with a zero-shot generation approach, using the`mistralai/Mistral-7B-Instruct-v0.2` model. The primary objective was to generate "Discharge Instructions" and "Brief Hospital Course" texts directly from the patient information in JSON format, without relying on fine-tuning or RAG techniques.

To guide the language model, we designed two ad-hoc prompt templates: one for "Discharge Instructions" and another for "Brief Hospital Course" summaries. These templates, created by us and not defined by medical professionals, included detailed instructions and placeholders for the patient JSON data. The "Discharge Instructions" template provided guidelines for generating a 300-400 word summary, covering aspects like greeting the patient, summarizing the hospital course, listing medications, and providing follow-up instructions. The "Brief Hospital Course" template aimed to produce a 400-600 word text, organized by active and inactive issues or organ systems, summarizing diagnostic findings, treatments, procedures, and the patient's response to treatment.

One notable limitation of using these ad-hoc prompt templates was the lack of grounding in external knowledge sources. The model relied solely on the information provided in the patient JSON, which may not always be comprehensive or sufficient for generating accurate and detailed summaries. Consequently, the generated summaries could sometimes miss important details, include irrelevant information, or lack the necessary context for certain medical terms or procedures.

To address these limitations and enhance the quality of the generated summaries, we explored

[1] The LLM is available at: `https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2`

[2] The library is available at: `https://github.com/noamgat/lm-format-enforcer`

Instruction: Generate a detailed "Brief Hospital Summary" in a structured format following a provided schema. The "Brief Hospital Summary" should include information about the patient's demographics, primary reason for admission, chief complaint, relevant past medical history, diagnostic findings, diagnosis, treatments provided, patient's response to treatment, consultations with other specialties, medication changes and adjustments, discharge condition and disposition, and follow-up plans, follow-up care instructions, and scheduled appointments.

Ensure that the extracted information is concise, clear, and captures the essential aspects of the patient's hospital course. Review the organized information for completeness and accuracy, and refine or rephrase any unclear or ambiguous information.

Schema: `patient_demographics`, `age`, `gender`, `name`, `reason_for_admission`, `chief_complaint`, `relevant_history`, `diagnosis`, `diagnostic_findings`, `imaging`, `labs`, `procedures`, `treatments`, `consultations`, `medications`, `medication_changes`, `discharge_condition`, `discharge_disposition`, `follow_up_instructions`.

Context:

[Clinical note(s) will be provided here]

Table 1: Patient Hospital Summary Prompt Template

RAG implementations, which are discussed in the subsequent section.

3.3 Retrieval Augmented Generation

To further enhance the LLM's generative capabilities, we sought to combine its parametric memory with non-parametric memory by enriching the prompt's context with relevant examples retrieved from an external dataset. Specifically, given an input $\mathbf{x}$ (a patient's JSON summary), we employ retrieval functions (defined later) to fetch the k most similar discharge instructions or brief hospital course texts from the Discharge Me training set $\mathcal{D}$. This process generates an k-shot prompt, thereby providing the LLM with additional context to inform its responses.

During the retrieval process, we calculate the relevance scores for all examples $d \in \mathcal{D}$ using two retrieval functions. For retrieval function A (BM25) (Robertson et al., 1995) the relevance score of a document d to a query x is calculated based on the frequency of query terms in the document, the document length, and the rarity of the query terms. For retrieval function B (SentenceBERT) (Reimers and Gurevych, 2019): the relevance score is computed as in expression (1).

$$s_B(\mathbf{x}, d) = \cos(\text{SentenceBERT}(\mathbf{x}), \text{SentenceBERT}(d)) \quad (1)$$

where SentenceBERT is a pre-trained model that encodes the input x and document d into dense vector representations. Specifically, we use the pre-trained `pritamdeka/S-PubMedBert-MS-MARCO` model (Deka et al., 2022)[3]. The cosine similarity between the two vector representations is used to measure the semantic similarity between the input and the document.

3.4 Results

We evaluated four different systems: FDS+SBERT-RAG, PS+Zero-shot, PS+SBERT-RAG, and PS+BM25-RAG. FDS+SBERT-RAG employed the full patient discharge summary (FDS) as input, along with the RAG framework and SentenceBERT (SBERT) for retrieval. PS+Zero-shot used the patient summary (PS) as input but performed inference using only the prompt instructions without RAG. PS+SBERT-RAG utilized the PS as input, also with the RAG framework and SBERT for retrieval. PS+BM25-RAG used the PS as input, with the RAG framework and BM25 as the retrieval function. Our top-performing system, PS+SBERT-RAG, attained an overall score of 0.183 at the competition deadline, exhibiting the potential of combining LLMs with RAG techniques for generating clinical notes. In contrast, our worst-performing system, PS+Zero-shot, obtained an overall score of 0.172, highlighting the performance uplift provided by our RAG methodology compared to the zero-shot approach. Table 2 presents our n-gram overlap metrics, table 3 our semantic similarity metrics, table 4 our factual alignment and clinical concept accuracy metrics, and table 5 our systems' overall scores.

[3] The model is available at: `https://huggingface.co/pritamdeka/S-PubMedBert-MS-MARCO`

System	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L
PS+Zero-shot	0.011	0.263	0.052	0.133
PS+SBERT-RAG	0.016	0.259	0.057	0.144
PS+BM25-RAG	0.018	0.244	0.051	0.141
FDS+SBERT-RAG	0.02	0.286	0.076	0.156

Table 2: N-gram Overlap Metrics

System	BERTScore	Meteor
PS+Zero-shot	0.238	0.275
PS+SBERT-RAG	0.282	0.284
PS+BM25-RAG	0.283	0.281
FDS+SBERT-RAG	0.261	0.290

Table 3: Semantic Similarity Metrics

System	AlignScore	MEDCON
PS+Zero-shot	0.210	0.196
PS+SBERT-RAG	0.210	0.215
PS+BM25-RAG	0.192	0.196
FDS+SBERT-RAG	0.170	0.219

Table 4: Factual Alignment and Clinical Concept Accuracy

In the context of n-gram overlap metrics, PS+SBERT-RAG exhibited suboptimal performance, achieving scores of 0.016 for BLEU-4, 0.259 for ROUGE-1, 0.057 for ROUGE-2, and 0.144 for ROUGE-L. These results suggest that the generated texts demonstrated limited lexical overlap with the reference summaries, implying potential challenges in accurately capturing relevant details and phrasing inherent in the gold standard.

On the other hand, PS+SBERT-RAG performed more favorably in semantic similarity metrics, achieving scores of 0.282 for BERTScore and 0.284 for Meteor. The BERTScore and Meteor results indicate that the generated texts from PS+SBERT-RAG exhibited high semantic equivalence with the reference summaries, suggesting its ability to capture the underlying meaning and context accurately, despite potential lexical differences.

Furthermore, PS+SBERT-RAG achieved a score of 0.21 for AlignScore, which evaluates the degree of factual alignment between the generated and reference texts. It also obtained a MEDCON score of 0.215, specifically gauging the accuracy and consistency of clinical concepts mentioned. These scores demonstrate the system's proficiency in generating clinically relevant and factually consistent content.

System	Overall
PS+Zero-shot	0.172
PS+SBERT-RAG	0.183
PS+BM25-RAG	0.175
FDS+SBERT-RAG	0.185

Table 5: Overall Evaluation Results

We also explored utilizing the full patient discharge summary (FDS) as input, along with the RAG framework, which we refer to as FDS+SBERT-RAG. We opted to use the Sentence-BERT retrieval function as it performed better than BM25 when using the PS inputs. Although we did not have the opportunity to finalize the results before the competition deadline, we found that FDS+SBERT-RAG achieved even better performance than PS+SBERT-RAG, with scores of 0.02 for BLEU-4, 0.286 for ROUGE-1, 0.076 for ROUGE-2, and 0.156 for ROUGE-L in the n-gram overlap metrics. FDS+SBERT-RAG also performed well in the semantic similarity metrics, scoring 0.261 for BERTScore, 0.29 for Meteor, 0.17 for AlignScore, and 0.219 for MEDCON. The improved performance of FDS+SBERT-RAG suggests that providing the model with more comprehensive patient information can further enhance its ability to generate accurate and clinically relevant summaries.

4 Conclusion

Our work explored a Retrieval-Augmented Generation approach for the "Discharge Me!" shared task on automating the generation of "Brief Hospital Course" and "Discharge Instructions" sections. We grounded a Large Language Model with structured patient summaries and retrieved relevant examples from the challenge training data set, aiming to mitigate hallucinations and enhance generation quality.

While our grounded approach demonstrated potential in generating coherent summaries, several areas exist for performance improvement. Fine-tuning the pipeline on "Brief Hospital Course" and "Discharge Instructions" sections could better align generated text with domain-specific language patterns. Incorporating constrained decoding or post-processing could improve n-gram overlap with references. Optimizing retrieval for stylistic similarity could indirectly benefit n-gram metrics. Moreover, metrics like MEDCON could be improved by retrieving Unified Medical Language System (UMLS) concept-rich examples or integrating UMLS databases during retrieval/generation. Exploring advanced RAG architectures with iterative retrieval and multi-step reasoning could address Naive RAG limitations.

Limitations

Our approach faced challenges due to maximum sequence length constraints. The retrieval encoder (SentenceBERT) had a 350-token limit, leading to the loss of relevant contextual information. Full discharge summaries exceeded the LLM's context length, resulting in omitted details, and likely hindered performance due to the loss of important contextual information. Additionally, our system did not effectively leverage the available radiology reports and ICD-9/10 diagnosis codes, which could potentially enhance the understanding of patient conditions and improve generation quality. The adhoc prompts, created without medical professionals' guidance, may have lacked necessary context and guidelines to generate accurate and comprehensive "Brief Hospital Course" and "Discharge Instructions" sections. The lack of domain adaptation for the LLM and SentenceBERT retrieval model could lead to issues understanding and generating domain-specific terminology and clinical concepts. By combining domain knowledge, task-specific fine-tuning, architectural enhancements, addressing sequence length limitations, and effectively integrating complementary data sources like radiology reports and diagnosis codes, we believe more accurate and reliable generation systems can be developed, contributing to improved patient care and reduced administrative burdens.

5 Acknowledgements

This work was elaborated within the framework of LOTU (TED2021-130398B-C22) funded by MCIN/AEI/10.13039/501100011033, European Commission (FEDER), and by the European Union “NextGenerationEU”/PRTR. Besides, this work was partially funded by the Spanish Ministry of Science and Innovation (EDHIA PID2022-136522OB-C22); by the Basque Government (IXA IT-1570-22) and by EXTEPA within Misiones Euskampus 2.0.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Pritam Deka, Anna Jurek-Loughrey, and P Deepak. 2022. Improved methods to aid unsupervised evidence-based fact checking for online health news. *Journal of Data Intelligence*, 3(4):474–504.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Xinbei Ma, Yeyun Gong, Pengcheng He, hai zhao, and Nan Duan. 2023. Query rewriting in retrieval-augmented large language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. `https://ai.meta.com/blog/meta-llama-3/`. Accessed: 2024-05-10.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, and Ilge Akkaya et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *Preprint*, arXiv:1908.10084.

Stephen Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. 1995. Okapi at trec-3. In *Overview of the Third Text REtrieval Conference (TREC-3)*, pages 109–126. Gaithersburg, MD: NIST.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and “discharge me!”. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Wen-Wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren’s song in the ai ocean: A survey on hallucination in large language models. *Preprint*, arXiv:2309.01219.

QUB-Cirdan at "Discharge Me!": Zero shot discharge letter generation by open-source LLM

Rui Guo[1,2] [*] **Greg Farnan**[2] [†] **Niall McLaughlin**[1] [‡] **Barry Devereux**[1] [§]
[1] Queen's University Belfast
[2] Cirdan

Abstract

The BioNLP ACL'24 Shared Task on Streamlining Discharge Documentation aims to reduce the administrative burden on clinicians by automating the creation of critical sections of patient discharge letters. This paper presents our approach using the Llama3 8B quantized model to generate the "Brief Hospital Course" and "Discharge Instructions" sections. We employ a zero-shot method combined with Retrieval-Augmented Generation (RAG) to produce concise, contextually accurate summaries. Our contributions include the development of a curated template-based approach to ensure reliability and consistency, as well as the integration of RAG for word count prediction. We also describe several unsuccessful experiments to provide insights into our pathway for the competition. Our results demonstrate the effectiveness and efficiency of our approach, achieving high scores across multiple evaluation metrics.

1 Introduction

The BioNLP ACL'24 Shared Task, "Discharge Me!" on Codabench (Xu et al., 2024), focuses on automating the creation of two crucial sections of patient discharge letters: "Brief Hospital Course" (BHC) and "Discharge Instructions" (DI). This initiative arises in response to significant time burdens on clinicians, highlighted by surveys of U.S. physicians. One study found that physicians spend twice as much time on Electronic Health Records (EHR) compared to direct patient interactions during clinical hours (Sinsky et al., 2016). Another survey involving 1,524 physicians revealed an average of 1.84 hours spent on EHR documentation outside office hours. Automating the generation of BHC and DI aims to significantly reduce the clerical load on healthcare providers, thereby improving patient service quality and potentially mitigating clinician burnout.

A discharge letter, or a discharge summary, is a critical document summarizing a patient's hospital visit from admission to discharge, serving as a bridge between hospital care and follow-up with outpatient providers. Among its several sections, the "Brief Hospital Course" outlines the patient's treatment and progress during the hospital stay, typically using clinical jargon best understood by healthcare professionals. Conversely, the "Discharge Instructions" are designed to guide patients and their caregivers once they leave the hospital, using layman's language to clearly explain follow-up care, medication regimens, and lifestyle recommendations.

Large Language Models (LLMs) offer a promising solution for automating medical documentation due to their ability to understand and generate human-like text (Singhal et al., 2023a; Zhang et al., 2023). Unlike traditional extractive summarization (El-Kassas et al., 2021), which predominantly involves concatenating snippets from existing texts, LLMs can enhance summarization by integrating both extractive and abstractive techniques. This has been applied to progress note summarization (Gao et al., 2022; Liu et al., 2023), similar to this Codabench challenge. With both proprietary LLMs such as ChatGPT (OpenAI, 2024) and open-source LLMs such as Llama3 (AI@Meta, 2024), the potential for creating accessible medical summaries is significant.

In this challenge, we propose a zero-shot approach utilizing the Llama3 8B quantized model, which is optimized for low computing resource usage without fine-tuning, and the result is in the top 10 in the final benchmark assessment. Our key contributions are:

[*] rui.guo@cirdan.com
[†] greg.farnan@cirdan.com
[‡] n.mclaughlin@qub.ac.uk
[§] b.devereux@qub.ac.uk

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 664–674
August 16, 2024. ©2024 Association for Computational Linguistics

- Crafting specialized templates for the “Brief Hospital Course” and “Discharge Instructions” sections, with carefully designed prompts to ensure the generated text is medically reliable and stylistically consistent with the training dataset.
- Exploring various methods to estimate the total word count for the target sections, including:
  - Fitting a statistical distribution
  - Employing a random forest classifier
  - Implementing a context-based retrieval system
- Conducting all experiments using a T4 GPU, demonstrating that our approach is computationally efficient.

2 Related Work

The application of foundation models, pre-trained on billions of tokens from diverse data sources, is increasingly prevalent in healthcare (He et al., 2024). These models are pivotal in various domains, such as diagnosis generation (Gao et al., 2023b) and medical image analysis (Zhang et al., 2024). Within clinical text processing, large language models (LLMs) are employed for tasks including summarization (Van Veen et al., 2023; Gao et al., 2023a) and answering medical questions (Singhal et al., 2023b). Specifically, the “Discharge me!” challenge involves condensing extensive medical records into succinct discharge letters while retaining all critical information, making LLMs suited for this task.

Participants in the BioNLP 2023 Workshop’s Problem List Summarization task often utilized T5 (Raffel et al., 2020) or BART (Lewis et al., 2019) models, enhancing these backbones either by further training on clinical texts or fine-tuning for specific clinical tasks (Gao et al., 2023a). This further pre-training introduces medical knowledge not originally present in the LLM while fine-tuning adapts the model to produce outputs in the correct format for the target task.

Several studies such as BioMistral (Labrak et al., 2024) and PMC-LLaMA (Wu et al., 2024) have adapted open-source LLMs by applying pre-training and fine-tuning sequentially. Conversely, Med-PaLM (Singhal et al., 2023a) bypasses additional pre-training, relying solely on fine-tuning from a vast pre-trained dataset. On a different note, BioMedLM (Bolton et al., 2024) focuses exclusively on medical texts, resulting in a smaller model but still competes effectively with models trained on larger, more general datasets.

Pre-training and fine-tuning LLMs require GPUs with significant memory capacities (often exceeding 16GB). Fine-tuning can take several days, even using Parameter-Efficient Fine Tuning (PEFT) methods like LoRA (Hu et al., 2021). However, modern LLMs can exhibit strong performance without additional fine-tuning if provided with the appropriate context and instructions. For instance, Almanac (Zakka et al., 2024) enhances its output by retrieving clinical question-related knowledge from curated sources, a technique known as Retrieval-augmented Generation (RAG) (Gao et al., 2023c). Additionally, Medagents (Tang et al., 2023) demonstrates that a zero-shot method, which deconstructs the question into distinct steps and assigns specific prompts and roles to the LLM for each stage, can achieve competitive results compared to more traditional few-shot approaches.

3 Methods

In this section, we introduce our zero-shot template-based approach, combined with RAG, to determine the target word count, which is both effective and resource-friendly. We adopted the Llama3 8B model with 8-bit quantization as the open-source model for this challenge. Figure 1 illustrates our approach:

1. Splitting the full discharge letter into different segments, such as “Chief Complaint” and “Brief Hospital Course”. This allows us to selectively use relevant sections and discard or truncate those too lengthy to process.
2. Employing Retrieval-Augmented Generation (RAG) to find the most similar patient’s target section, using that section’s word count as the target for generation. Generating a similar word count to the target can help maintain the generated summaries’ completeness and increase evaluation metrics such as BLEU, ROUGE, and METEOR.
3. Providing the target section’s structure template and prompt to Llama3 along with

the patient's context and target word count.

4. Generating the result by Llama3 8B quantized model.

While GPT-4/3.5 models generally outperform open-source models such as Llama2 in understanding EHR data (Liu et al., 2024), the rules of this challenge discourage the use of proprietary model APIs (e.g., OpenAI's GPT-4). Consequently, we resorted to the state-of-the-art (SOTA) open-source model, Llama3 (AI@Meta, 2024). Our approach leverages the full text from the "text" field in the provided discharge.csv file, alongside aggregated fields from other MIMIC-IV tables, including patient information, diagnoses, and transfer history. We meticulously curated a template for each target section and designed prompts to guide the LLM in generating the required sections. In addition to our final approach, we documented several other zero-shot methods for target section generation and various approaches to predict the target section's word count. However, these were not adopted in our final solution.

3.1 Dataset Exploration

The dataset for this challenge is derived from MIMIC-IV's submodules, MIMIC-IV-Note (Johnson et al., 2023c) and MIMIC-IV-ED (Johnson et al., 2023a). All patients have visited the Emergency Department (ED), and the final target sections, "Brief Hospital Course" and "Discharge Instructions", are extracted from their discharge letters. Since patients can be admitted to the hospital after their initial ED visit, we also explored other tables from the MIMIC-IV hosp and ICU modules (Johnson et al., 2023b) to provide a comprehensive view of the patient's hospital stay beyond the ED information.

Due to limited context length, we could not simply pass all available information into the LLM. Therefore, we ranked all sections of the discharge letter to select a subset of the information. We segmented the discharge letter's "text" column from discharge.csv using regex and a template of keywords for different sections, as shown in the Section column of Table 1. Besides the information from the "text" column, we aggregated "Patient Admissions" information, including gender, race, age (calculated), "Diagnoses" (throughout the patient stay), and "Transfer Summary" from other MIMIC-IV tables. Since we compiled the patient's diagnoses and transfer summary for the entire hospital stay using other MIMIC-IV tables rather than just the Emergency Department (ED) stay, we did not use the tables in the ED module, such as triage, edstays, and diagnosis, as they only cover part of the patient's stay. The content of "radiology" will be set to the content of the section "Imaging" if the "Imaging" section is empty in the discharge letter. We then calculated the average ranking of the metric score for each section relative to the target sections, using the provided evaluation metrics, including BLEU-4 (Papineni et al., 2002), ROUGE-1/2/L (Lin, 2004), BERTScore (Zhang et al., 2019), Meteor (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023). Each section was compared to the target sections, "Brief Hospital Course" (BHC) and "Discharge Instructions" (DI), with higher-ranking sections being more related to the target sections. Table 1 shows that "History of Present Illness" is most related to the BHC section, followed by imaging results, physical exams, past medical history, and diagnoses. BHC is most related to DI, followed by sections related to BHC.

Based on the ranking in Table 1 and the length of each section, we selected "History of Present Illness", "Imaging and Studies", "Past Medical History", "Patient Admissions", and "Chief Complaint" as the context for the BHC section. We used the generated BHC, "Discharge Medications", "Discharge Disposition", "Discharge Diagnoses", "Discharge Condition", and "Followup Instructions" for DI section. Other sections related to DI were excluded because they are also related to BHC. We truncated each section to the 95th percentile of its total length to remove outliers and potential segmentation errors.

3.2 Retrieval for the Target Section Word Count

Understanding the target section's word count is beneficial for generating the appropriate amount of text, thereby improving the evaluation metrics for this challenge. Figure 2 shows the word count distribution for the target sections in the training dataset. Both target sections have right-skewed distributions, and BHC also has a peak for word counts under 100. We hypothesize that patients with similar backgrounds may have similar target sections. These retrieved target sections from patients with similar backgrounds can be used

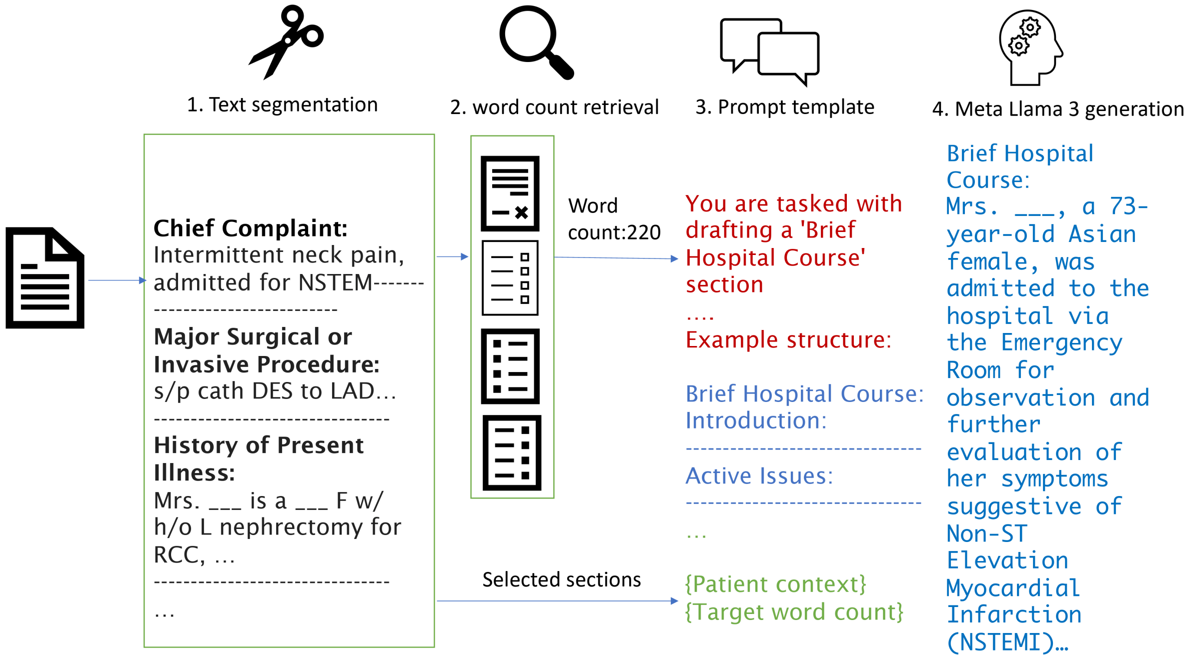

1. Text segmentation
2. word count retrieval
3. Prompt template
4. Meta Llama 3 generation
Chief Complaint: Intermittent neck pain, admitted for NSTEM-------
Major Surgical or Invasive Procedure: s/p cath DES to LAD…
History of Present Illness: Mrs. ___ is a ___ F w/ h/o L nephrectomy for RCC, …
…
Word count:220
Selected sections
You are tasked with drafting a 'Brief Hospital Course' section
….
Example structure:
Brief Hospital Course: Introduction:
Active Issues:
…
{Patient context}
{Target word count}
Brief Hospital Course:
Mrs. ___, a 73-year-old Asian female, was admitted to the hospital via the Emergency Room for observation and further evaluation of her symptoms suggestive of Non-ST Elevation Myocardial Infarction (NSTEMI)…

Figure 1: Overview of our solution. The figure illustrates our four-step approach: (1) Text Segmentation: splitting the discharge letter into sections such as "Chief Complaint" and "Brief Hospital Course"; (2) Retrieval-Augmented Generation (RAG): retrieving similar patient sections to determine word count; (3) Template and Prompt Design: providing structured templates and prompts to Llama3 with patient context and target word count; (4) Text Generation: generating the final output using Llama3.

as a starting point, providing a template or word count for further refinement. We selected "Chief Complaint", "Diagnoses", and "History of Present Illness" as inputs for retrieving the BHC section. We added "Admission medications", "Discharge Medications", "Discharge Disposition", "Discharge Diagnoses", and "Discharge Condition" for retrieving the DI section. We used the "sentence-transformers/all-MiniLM-L6-v2" model to create embeddings of the context information for each training dataset entry and FAISS for similarity search. The word count from the first retrieved document's target section was used in the prompt to LLM for the generation. We compared this word count selection strategy to using a fixed word count, and the results are presented in Section 4.

3.3 Target Section Structure Template and Prompt Creation

The target word count distribution varies, and we inspected several randomly chosen examples of target sections with different word counts. We selected examples with word counts over 180 to accommodate most cases for BHC template construction. Examples with word counts between 100-300 were chosen for the DI template construction. The structure is in JSON format, with names and descriptions for each section.

The BHC structure template is:

```
1. Introduction: Brief introduction
   including patient demographics,
   significant past medical history,
   and reason for hospitalization.

2. Active Issues: Details of the primary
   medical concerns addressed during the
   stay, including initial assessments
   and management actions.

3. Chronic Issues (Optional):
   Management of known chronic
   conditions during the hospital
   stay.

4. Transitional Issues (Optional):
   Specific follow-up actions
   recommended for post-discharge
   care.

5. Additional Notes (Optional):
   Other pertinent information or
```

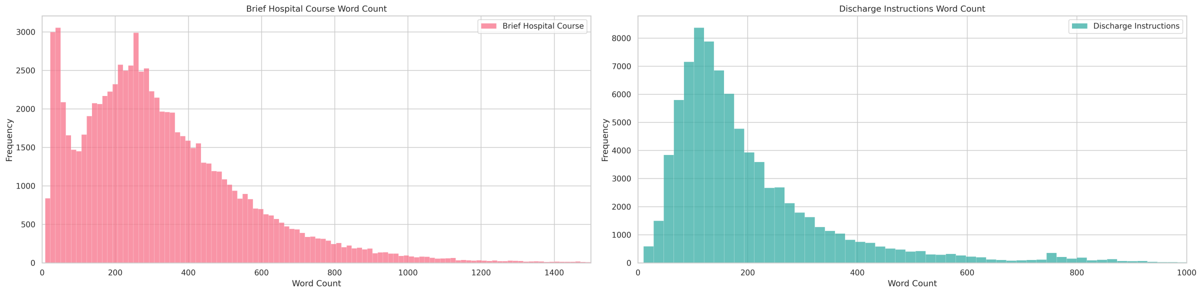


Figure 2: The target section word count distribution. Both BHC and DI have right-skewed distributions. BHC has two peaks, one below 100 words and one around 250 words.

```
   considerations affecting patient
   care.
```

Section	BHC	DI
Patient Admissions	13	21
Transfer Summary	15	23
Diagnoses	5	4
Service	11	12
Allergies	14	22
Attending	17	24
Chief Complaint	8	11
Major Surgical Procedure	9	17
History of Present Illness	1	2
Review of System	10	15
Past Medical History	4	9
Social History	16	25
Family History	12	16
Physical Exam	3	5
Pertinent Results	7	18
Imaging and Studies	2	3
Brief hospital course		1
Admission Medications		10
Discharge Medications		7
Discharge Disposition		14
Discharge Diagnoses		6
Discharge Condition		8
Followup Instructions		13
Provider		19
Code Status		20

Table 1: The ranking of different sections' relation to BHC/DI by averaging all the evaluation metrics provided by this challenge. We aggregated the patient's admission info, including gender, race, age (calculated), diagnosis, and transfer history from other MIMIC-IV tables.

The template includes several optional sections not included in all the examples. The template will be fed to the prompt below as the "structure" variable. The prompt for BHC is:

```
As a medical professional, you are
tasked with drafting a “Brief Hospital
Course” section for a discharge letter.
Utilize the structure from a brief
hospital course example to guide your
composition.  The goal is to write a
new, coherent, brief hospital course for
another patient based on the provided
structured template.  The total word
count for the brief hospital course
should be {words} words.

BHC Instructions:

 1. Follow the JSON template provided
    to structure the new brief hospital
    course. Each section should be filled
    according to the relevant patient
    information.

 2. Omit the optional sections if they are
    irrelevant to the patient’s case.

 3. Omit the optional sections if the
    total word count is less than 100
    words.

 4. Do not add a new section after
    Additional Notes.

 5. Use placeholders “___” for any date,
    patient name, and location.

 6. Use appropriate medical terminology
    and concise language to ensure
```

clarity and professionalism.

7. Do not be wordy; be concise if possible.

8. Do not include the word "optional" in the result if they are included. If they are not included, just omit those sections.

9. Do not copy patient information verbatim; paraphrase and use the structure template to fit in the details.

10. All the section headers must be from the template, not from the patient information.

11. Do not fabricate details not present in the patient information.

12. Use section headers for each major medical issue, starting with a hashtag #, do not use * for section header.

13. Use bullet points to highlight key actions, medication changes, or critical clinical decisions, starting with a hyphen -. Do not use * or +.

14. Ensure that each major issue or condition has its own section header if there is enough content related to it, even if briefly mentioned.

15. Write in a narrative style for each section, providing a detailed account of the patient's condition, treatment, and outcomes.

16. Employ medical abbreviations and terminology appropriately to convey information efficiently.

17. Start the output with "Brief hospital course:"

Example structure for the brief hospital course: {structure}.
Patient information: {context}.

The template for DI is below. This is fed to the DI prompt as the "structure" variable.

1. Greeting: "Dear [Title] ___,", "HospitalExperience": "It was a pleasure taking care of you at ___.",

2. AdmissionReason: "Title": "WHY WAS I ADMITTED TO THE HOSPITAL?", "Details": "[ReasonForAdmission]" ,

3. InHospitalActivities: "Title": "WHAT HAPPENED WHILE I WAS IN THE HOSPITAL?", "Details": "[ActivitiesDuringStay]" ,

4. DischargeAdvice: "Title": "WHAT SHOULD I DO WHEN I GO HOME?", "Instructions": "[PostDischargeInstructions]" ,

5. Closing: "We wish you the best!", "CareTeam": "Your ___ Team"

The prompt for DI is:

You are tasked with drafting a "Discharge Instructions" section for a patient's discharge letter as a medical professional. The instructions should succinctly summarize the key points of the patient's hospital stay and post-discharge care clearly and easily for the patient to follow.

DI Instructions:

1. Use the JSON template provided to structure the discharge instructions.

2. Do not include explicit section headers in the final text, such as "Greeting" or "Hospital Experience".

3. Do not include any placeholder such as "[]" in the result.

4. Include the title in the template.

5. Integrate medication information narratively, mentioning specific medications only when discussing their relevance to the patient's ongoing care and follow-up instructions.

6. Do not list medications; describe how they contribute to the patient's treatment plan.

7. The total word count should be around {words} words, focusing on essential instructions relevant to the patient's care.

8. Use "___" to anonymize any date, patient name, and location.

9. Clearly specify any medication changes, follow-up appointments, and additional care instructions using placeholders where specific details are to be inserted.

10. Employ a professional yet empathetic tone to ensure clarity and approachability.

11. Integrate medical terminology appropriately, ensuring it is understandable to a layperson.

12. Start the output with a polite greeting and conclude with well-wishes or a thank you message.

Example structure for the discharge instructions: {structure}.
Patient information: {context}.

4 Results

The Llama3 model was downloaded from the Ollama model repository with the model ID "llama3:8b-instruct-q8_0". We utilized the LangChain framework for retrieval, template building, and model calling. All experiments were conducted on a T4 GPU with 16GB memory, using the Microsoft Azure platform's "Standard NC4 as T4 v3 (4 vCPUs, 28 GiB memory)" configuration.

We compared several approaches:

1. *Baseline with Random Shuffling*: We shuffled the "hadm_id" column, a unique identifier for each patient's discharge letter, assigning a random target section to each "hadm_id". This random selection comes from the same distribution as the training data but without the actual content of the input text.

2. *Baseline with RAG Retrieval*: We used the retrieved target sections directly. This result can be similar to the target, but the details can differ from the real input.

3. *Fixed Target Word Count*: We set a fixed word count of 420 for BHC and 100-200 for DI in the prompt.

4. *Proposed Method*: Our method combines retrieved target word counts with a structured template.

Table 2 presents the evaluation metrics from the Codabench platform (Xu et al., 2024), including BLEU-4 (Papineni et al., 2002), ROUGE-1/2/L (Lin, 2004), BERTScore (Zhang et al., 2019), Meteor (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023). The random shuffle yielded the lowest scores across all metrics, indicating poor performance. Using the retrieved target section directly resulted in the highest BLEU score. The fixed word count approach achieved higher Align and MEDCON scores than the retrieved target section but had lower scores for other metrics. Our proposed method, which combines the retrieved word count and structured template, achieved the highest scores across all metrics except BLEU. The lower BLEU score for the proposed method is due to BLEU's heavy penalty for deviations from exact wording. In contrast, the higher ROUGE scores indicate our method effectively captures the essential content, even with varied wording. We also measured the generation time for each section. The average time to generate one BHC was 16.67 seconds, and one DI was 16 seconds.

5 Unsuccessful Attempts

We also explored several alternative approaches for this task, but they yielded unsatisfactory results:

1. *Style Transfer Using Retrieved Target Section*: We asked the LLM to use the style of the retrieved target section to fit the patient context. However, the Llama3 8B model often used the target section directly, failing to infer the style and remove the original content. This could be due to the weaker reasoning ability of the 8B model compared to the 70B model with better reasoning ability.

2. *Two-Step Style Transfer*:

   (a) Firstly, extract a template from the target section.

   (b) Secondly, fill in the patient content into the template (this step can also be split into several smaller steps).

	bleu	rouge1	rouge2	rougel	bertscore	meteor	align	medcon	overall
random shuffle	0.01	0.183	0.025	0.105	0.226	0.23	0.109	0.1	0.124
RAG retrieved target	**0.041**	0.286	0.061	0.172	0.293	0.297	0.167	0.203	0.19
fixed target word	0.017	0.296	0.055	0.159	0.256	0.285	0.187	0.221	0.185
retrieved word count	0.024	**0.377**	**0.106**	**0.205**	**0.3**	**0.332**	**0.174**	**0.254**	**0.221**

Table 2: The evaluation results from the Codabench platform. The random shuffle method yielded the lowest scores, while our final retrieval approach to determine the target word count achieved the highest scores across most metrics.

However, the extracted templates were not always reliable, and this method took twice as long as the curated template approach. Consequently, we opted to curate the templates rather than relying on the LLM manually.

3. *Predicting Target Section Word Count*: We tested several methods to predict the total word count of the target section, including fitting a random forest classifier by aggregating over 100 features from other MIMIC-IV tables and fitting log-normal distributions. These methods also proved inadequate. Table 3 shows the random forest classifier results for BHC with word count classes greater than 450, with an F1 score of 0.45. Figure 3 lists the top 10 features, including the number of lab tests, diagnoses, and total hospital duration. The classifier achieved an F1 score of 0.49 for word counts greater than 280 for the DI section, as shown in Table 4, with different section word counts being the top features in Figure 4.

	precision	recall	f1-score	support
<450	0.818	0.926	0.869	18965
>450	0.610	0.359	0.452	6087

Table 3: BHC random forest classifier results for BHC word count above and below 450. The f1-score is 0.45 for the class with more than 450 words, which is not accurate enough.

6 Conclusion

In this paper, we present a resource-friendly approach to automating the generation of the “Brief Hospital Course” and “Discharge Instructions” sections in discharge letters using the Llama3 8B quantized model. Our zero-shot template-based method and Retrieval-Augmented Generation produce high-quality, contextually appropriate

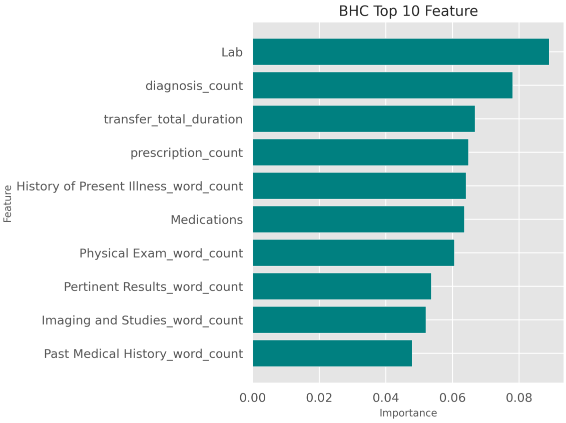


Figure 3: The top 10 features for the BHC classifier. WC: word count. The total number of lab tests, diagnosis, and total duration in the hospital are the top 3 features.

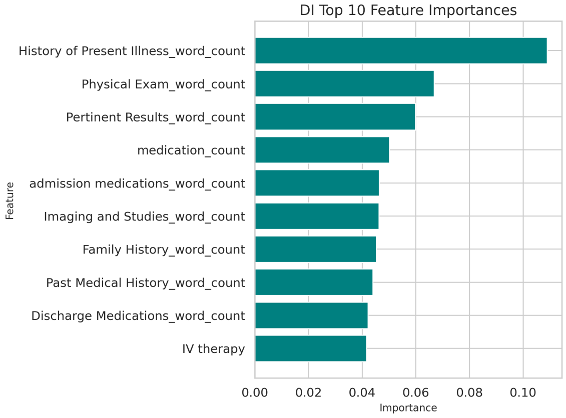


Figure 4: The top 10 features for the DI classifier. WC: word count. The word count of different segments is ranking high.

	precision	recall	f1-score	support
<280	0.864	0.964	0.911	20143
>280	0.716	0.377	0.494	4909

Table 4: DI random forest classifier result for DI word count above and below 280. The f1-score is 0.49 for the class with more than 280 words, which is not accurate enough.

summaries. However, we observe a lower BLEU score due to the different wording between the method's result and the target sections. Ensuring the reliability and accuracy of generated content remains a significant challenge. Future work will focus on enhancing model reasoning capabilities, improving dynamic template extraction, and integrating robust validation mechanisms to verify medical accuracy. The code for this work is shared on https://github.com/ruiguo-bio/discharge_me, covering aggregating additional tables, segmentation of the discharge letters, RAG for the two target sections, and the random forest classifier for the target section words prediction.

7 Limitations and Future Work

1. We would like to perform a more thorough evaluation to ensure that the model's generated content is clinically relevant and does not include false or harmful information. This evaluation could be extended to understanding the strengths and weaknesses of language models for the challenge task.

2. We create a template by sampling target sections with word counts close to the median. However, the length and structure of real target sections can vary significantly from our template. Our approach could be improved by predicting the target word count more precisely or by sampling different templates depending on the word count.

3. We would like to test a wider range of language models and thoroughly compare different methods of providing relevant context to the language model, including different methods of Retrieval-Augmented Generation (RAG) and prompt engineering.

8 Ethical Statement

All the data used in the experiments are downloaded from the PhysioNet after completing the required CITI training and credentialing process. Beyond the general potential ethical considerations of using LLMs to automatically process and generate clinical text (including bias, fairness, transparency and accountability), there are no specific ethical issues raised by the particular methodologies or data presented in this research.

References

AI@Meta. 2024. Llama 3 model card. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md. Accessed: 2024-05-12.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Elliot Bolton, Abhinav Venigalla, Michihiro Yasunaga, David Hall, Betty Xiong, Tony Lee, Roxana Daneshjou, Jonathan Frankle, Percy Liang, Michael Carbin, et al. 2024. Biomedlm: A 2.7 b parameter language model trained on biomedical text. *arXiv preprint arXiv:2403.18421*.

Wafaa S El-Kassas, Cherif R Salama, Ahmed A Rafea, and Hoda K Mohamed. 2021. Automatic text summarization: A comprehensive survey. *Expert systems with applications*, 165:113679.

Yanjun Gao, Dmitriy Dligach, Timothy Miller, Matthew M Churpek, and Majid Afshar. 2023a. Overview of the problem list summarization (probsum) 2023 shared task on summarizing patients' active diagnoses and problems from electronic health record progress notes. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2023, page 461. NIH Public Access.

Yanjun Gao, Ruizhe Li, John Caskey, Dmitriy Dligach, Timothy Miller, Matthew M Churpek, and Majid Afshar. 2023b. Leveraging a medical knowledge graph into large language models for diagnosis prediction. *arXiv preprint arXiv:2308.14321*.

Yanjun Gao, Timothy Miller, Dongfang Xu, Dmitriy Dligach, Matthew M Churpek, and Majid Afshar. 2022. Summarizing patients' problems from hospital progress notes using pre-trained sequence-to-sequence models. In *Proceedings of COLING. International Conference on Computational Linguistics*, volume 2022, page 2979. NIH Public Access.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023c. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Yuting He, Fuxiang Huang, Xinrui Jiang, Yuxiang Nie, Minghao Wang, Jiguang Wang, and Hao Chen. 2024.

Foundation model for advancing healthcare: Challenges, opportunities, and future directions. *arXiv preprint arXiv:2404.03264*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Leo Anthony Celi, Roger Mark, and Steven Horng. 2023a. Mimic-iv-ed (version 2.2).

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. 2023b. Mimic-iv (version 2.2).

Alistair Johnson, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. 2023c. Mimic-iv-note: Deidentified free-text clinical notes (version 2.2).

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. Biomistral: A collection of open-source pretrained large language models for medical domains. *arXiv preprint arXiv:2402.10373*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Darren Liu, Cheng Ding, Delgersuren Bold, Monique Bouvier, Jiaying Lu, Benjamin Shickel, Craig S Jabaley, Wenhui Zhang, Soojin Park, Michael J Young, et al. 2024. Evaluation of general large language models in contextually assessing semantic concepts extracted from adult critical care electronic health record notes. *arXiv preprint arXiv:2401.13588*.

Ming Liu, Dan Zhang, Weicong Tan, and He Zhang. 2023. Deakinnlp at probsum 2023: Clinical progress note summarization with rules and language modelsclinical progress note summarization with rules and languague models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 491–496.

OpenAI. 2024. Chatgpt. `https://openai.com/chatgpt`. Accessed: 2024-05-12.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. 2023a. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, et al. 2023b. Towards expert-level medical question answering with large language models. *arXiv preprint arXiv:2305.09617*.

Christine Sinsky, Lacey Colligan, Ling Li, Mirela Prgomet, Sam Reynolds, Lindsey Goeders, Johanna Westbrook, Michael Tutty, and George Blike. 2016. Allocation of physician time in ambulatory practice: a time and motion study in 4 specialties. *Annals of internal medicine*, 165(11):753–760.

Xiangru Tang, Anni Zou, Zhuosheng Zhang, Yilun Zhao, Xingyao Zhang, Arman Cohan, and Mark Gerstein. 2023. Medagents: Large language models as collaborators for zero-shot medical reasoning. *arXiv preprint arXiv:2311.10537*.

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, Eduardo Pontes Reis, Anna Seehofnerova, et al. 2023. Clinical text summarization: Adapting large language models can outperform human experts. *Research Square*.

Chaoyi Wu, Weixiong Lin, Xiaoman Zhang, Ya Zhang, Weidi Xie, and Yanfeng Wang. 2024. Pmc-llama: toward building open-source language models for medicine. *Journal of the American Medical Informatics Association*, page ocae045.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10(1):586.

Cyril Zakka, Rohan Shad, Akash Chaurasia, Alex R. Dalal, Jennifer L. Kim, Michael Moor, Robyn Fong, Curran Phillips, Kevin Alexander, Euan Ashley, Jack Boyd, Kathleen Boyd, Karen Hirsch, Curt Langlotz, Rita Lee, Joanna Melia, Joanna Nelson, Karim Sallam, Stacey Tullis, Melissa Ann Vogelsong, John Patrick Cunningham, and William Hiesinger. 2024. Almanac — retrieval-augmented language models for clinical medicine. *NEJM AI*, 1(2):AIoa2300068.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Hongbo Zhang, Junying Chen, Feng Jiang, Fei Yu, Zhihong Chen, Guiming Chen, Jianquan Li, Xiangbo Wu, Zhang Zhiyi, Qingying Xiao, Xiang Wan, Benyou Wang, and Haizhou Li. 2023. HuatuoGPT, towards taming language model to be a doctor. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10859–10885, Singapore. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Zhenyu Zhang, Benlu Wang, Weijie Liang, Yizhi Li, Xuechen Guo, Guanhong Wang, Shiyan Li, and Gaoang Wang. 2024. Sam-guided enhanced fine-grained encoding with mixed semantic learning for medical image captioning. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1731–1735. IEEE.

e-Health CSIRO at “Discharge Me!” 2024: Generating Discharge Summary Sections with Fine-tuned Language Models

Jinghui Liu, Aaron Nicolson, Jason Dowling, Bevan Koopman, & Anthony Nguyen
Australian e-Health Research Centre, CSIRO, Brisbane, Australia
jinghui.liu@csiro.au

Abstract

Clinical documentation is an important aspect of clinicians’ daily work and often demands a significant amount of time. The BioNLP 2024 Shared Task on Streamlining Discharge Documentation (Discharge Me!) aims to alleviate this documentation burden by automatically generating discharge summary sections, including brief hospital course and discharge instruction, which are often time-consuming to synthesize and write manually. We approach the generation task by fine-tuning multiple open-sourced language models (LMs), including both decoder-only and encoder-decoder LMs, with various configurations on input context. We also examine different setups for decoding algorithms, model ensembling or merging, and model specialization. Our results show that conditioning on the content of discharge summary prior to the target sections is effective for the generation task. Furthermore, we find that smaller encoder-decoder LMs can work as well or even slightly better than larger decoder-based LMs fine-tuned through LoRA. The model checkpoints from our team (**aehrc**) are openly available.[1]

1 Introduction

Clinical documentation in the age of Electronic Health Records (EHRs) can be a significant burden to clinicians in recording clinical information effectively (Colicchio et al., 2020; Rule et al., 2021). This reduces the time clinicians spend interacting with their patients and could lead to stress and burnout (Colicchio et al., 2019), degrading both the quality of patient care and the experience of care providers (Shanafelt et al., 2016).

Language Models (LMs) have demonstrated impressive NLP capabilities and are considered to have the potential to reduce the clinical documentation burden by automatically generating clinical text (Patel and Lam, 2023; Roberts, 2024; Omiye et al., 2024). For example, a recent study (Van Veen et al., 2024) demonstrated that LMs can generate succinct clinical summaries from text including progress notes and patient-doctor dialogues, sometimes even preferred over those written by medical experts. The BioNLP 2024 Shared Task “Discharge Me!” (Xu et al., 2024) focuses on generating the discharge summary (or discharge note) to assess the potential of LMs for this specific type of clinical note, which is often more time-consuming for clinicians to document and also more challenging to model given its length and complexity.

This paper presents the submissions from e-Health CSIRO in the shared task. We approach the task by fine-tuning multiple open-sourced LMs, including both decoder-only and encoder-decoder models. We fine-tune these models to generate two specific sections from discharge notes: *brief hospital course* and *discharge instruction*, by conditioning on the prior content in the notes as context. We explore various configurations with input context, decoding, ensembling, and target specialization. We find that much smaller encoder-decoder LMs could have a slight edge over fine-tuning decoder-only LMs (all with the size of 7/8B parameters) with LoRA (Hu et al., 2022). Our best submission ranked 3^{rd} on the final leaderboard under both automatic and manual evaluation.

2 Methods

2.1 Task and Dataset

The Shared Task focuses on generating two important sections of discharge notes: brief hospital course (BHC) and discharge instruction (DI). The first section provides a snapshot of the important information about the patient care during the hospital, and the second a summary to communicate that information and instructions after leaving the hospital to patients. The audiences for the two

[1] https://github.com/JHLiu7/bionlp24-shared-task-discharge-me

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 675–684
August 16, 2024. ©2024 Association for Computational Linguistics

sections are different as the former is read by clinicians while the latter by patients. The Shared Task uses the MIMIC-IV database (Johnson et al., 2023) to curate the dataset consisting of 109,168 patients, which are split into Train (68,785), Validation (14,719), Phase I testing (14,702), and Phase II testing (10,962). Each patient has a discharge summary that includes both sections, and participants are allowed to utilize data elements in the EHR database beyond the note alone as input.

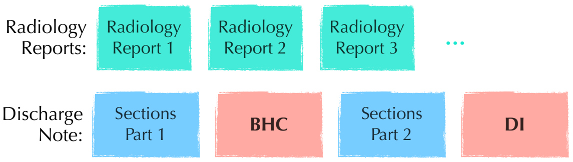


Figure 1: Illustration of the contents in clinical notes.

Our experiments focus only on the free-text clinical notes as input and do not consider other data modalities. We primarily use the content in the discharge note prior to the corresponding target section as input context. Radiology reports are considered optionally. We depict the note structures in Figure 1. Specifically, we consider the base context for BHC as C^{bhc}_{base} = *"Sections Part 1"*, and for DI as C^{di}_{base} = *"BHC"* + *"Sections Part 2"*. We consider two types of prolonged contexts: 1) $C_{base+rad} = C_{base}$ + *"Rad Reports"*, where radiology reports are concatenated with with the related sections; and 2) C^{di}_{long} = *"Sections Part 1"* + C^{di}_{base}, which extends the input context for DI. We then train models to generate the target sections T^{bhc} and T^{di} based on the corresponding contexts.

2.2 Language Models

We consider both decoder-only and encoder-decoder LMs for our experiments. For decoder-only LMs, we examine three popular open-sourced models at 7/8 billion paramater levels, including Llama3-8B [2], Mistral-7B (Jiang et al., 2023), and Gemma-7B (Gemma Team, 2024), all based on the instruction-tuned versions, denoted as Llama3-it, Mistral-it, and Gemma-it. Additionally, we examine the base version of Llama3-8B, denoted simply as Llama3. For encoder-decoder LMs, we focus on PRIMERA (447M) (Xiao et al., 2022) and Long-T5 (770M, global attention) (Guo et al., 2022), both capable of handling long input and output lengths.

	# Max Tokens (Llama3)	# Max Tokens (PRIMERA)
C^{bhc}_{base}	2816	3328
C^{di}_{base}	2048	2048
C^{di}_{long}	4608	5120
$C^{bhc}_{base+rad}$	4608	5120
$C^{di}_{base+rad}$	3840	4096
T^{bhc}	1280	1280
T^{di}	512	512

Table 1: Number of maximum tokens for modeling.

To determine the maximum lengths for modeling, we calculate the 85th percentile of the number of tokens and round it up to a multiplier of 256 for each LM. We present the statistics for Llama-3 and PRIMERA in Table 1 as examples. With each LM, we train two independent models for BHC and DI. For decoder-only LMs, we construct the prompt template similar to Alpaca (Taori et al., 2023), shown in Appendix Figure 2.

We then fine-tune these LMs for the text generation task. The decoder-only LMs, on the other hand, are loaded in half-precision (BF16) and fine-tuned through LoRA. We follow the setup from Dettmers et al. (2023) and use lr = 2e-4, r = 64, $alpha$ = 16, with LoRA attached to all linear layers. The encoder-decoder LMs are fully fine-tuned with lr = 5e-5. All LMs are trained with batch size of 16 for 5 epochs using Adam, with 3% ratio for linear warmup. We use the default generation configuration, including the decoding algorithms, for the pretrained LMs. All experiments are performed on NVIDIA H100 GPU.

2.3 Evaluation

The automatic evaluation is based on 8 popular pairwise metrics, including BLEU-4 (Papineni et al., 2002), ROUGE-1/2/L (Lin, 2004), BERTScore (Zhang* et al., 2020), Meteor (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023). They present a diverse set of measurements for string overlaps, semantic similarity, and medical concept mapping. The results for BHC and DI are averaged for each metric. The final ranking of the Shared Task is based on the average of the all scores on 250 hidden cases from Phase II testing, although participants are required to submit generation for all cases.

2.4 Experimental Setup

We investigate several factors that could impact the generation performance and compare them with

[2] https://ai.meta.com/blog/meta-llama-3/

Model	Overall	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	Meteor	AlignScore	MEDCON
Fine-tuning baselines based on C_{base}									
Llama3	28.05	10.05	35.65	13.56	25.65	38.66	39.98	25.93	34.90
Llama3-it	23.53	7.88	25.56	9.66	15.70	35.13	38.90	22.73	32.69
Mistral-it	23.71	5.46	32.43	12.23	21.04	30.58	34.49	23.11	30.34
Gemma-it	25.14	6.31	35.04	11.18	24.53	32.91	36.07	23.46	31.60
PRIMERA	29.17	10.55	40.33	15.94	25.69	41.17	37.92	26.49	35.28
Long-T5	22.47	6.31	30.16	8.88	19.12	32.31	31.44	22.50	29.07
Extended Input Context									
Llama3 w/ $C_{base+rad}$	25.15	8.69	27.24	10.81	19.20	37.26	39.17	25.11	33.71
PRIMERA w/ $C_{base+rad}$	29.10	10.64	39.76	15.75	27.10	40.31	37.55	27.10	34.61
Llama3 w/ C^{di}_{long}	28.33	9.56	37.27	12.93	25.87	38.67	40.64	26.67	35.04
PRIMERA w/ C^{di}_{long}	28.26	10.14	38.93	13.48	23.73	40.68	37.95	26.80	34.37
Unified LM for both T_{bhc} and T_{di}									
Llama3 (single)	25.38	7.89	31.79	11.34	21.89	35.38	38.91	23.39	32.42
Alternative Decoding for Llama3									
Llama3 w/ beam	25.20	10.06	29.14	8.05	17.83	37.05	40.57	26.17	32.71
Llama3 w/ constrastive	24.09	8.36	27.81	10.13	18.24	36.10	35.52	26.37	30.21
Ensemble Decoding									
Llama3 + Llama3	26.17	9.67	28.79	11.36	21.02	38.31	39.71	26.29	34.24
Llama3 + Llama3-it	27.04	9.66	32.68	12.99	22.30	37.96	39.79	26.10	34.84
Merging LoRA Adapters									
Llama3 x2 LoRA	25.78	8.20	34.48	11.60	22.69	35.81	37.44	23.25	32.79
Llama3 x4 LoRA	21.80	4.50	33.05	11.97	20.45	30.79	28.31	17.35	28.00

Table 2: Results from automatic evaluation, based on 250 hidden samples from Phase II testing.

the base generation setup, in which two LMs of the same architecture are trained on C_{base} for BHC and DI, respectively. We examine the impact of extended input context by replacing C_{base} with $C_{base+rad}$ or C^{di}_{base} with C^{di}_{long}. Taking Llama3 as the example, we explore a variety of modifications, including training a unified LM that models the two targets jointly to explore the benefit of target specialization. We also apply various decoding algorithms other than greedy search, including beam search ($n = 4$), and contrastive search ($\alpha = 0.6$, $k = 6$) (Su et al., 2022). Furthermore, we explore ensemble decoding (Manakul et al., 2023) and the popular adapter merging with Llama3 as the example. The former averages the logits from two LMs for generating each token with greedy search, and the latter applied TIES (Yadav et al., 2023) to merge the paramters of several LoRA adapters (equal weights, density of 0.5) before attaching it to the main LM. Finally, we prompt instruction-tuned LMs in the zero-shot manner, including the 70B checkpoints, on a subset of validation to observe the benefit of fine-tuning for this task.

3 Results & Analysis

3.1 Both Decoder and Encoder-encoder LMs Work Well When Fine-tuned

We firstly find all LMs obtain decent results when fine-tuned with C_{base}. Meanwhile, the instruction-tuned decoder-only LMs perform worse than the base version of Llama3. This aligns with existing findings that instruction tuning could harm performance on NLP benchmarks (Ouyang et al., 2022; Ivison et al., 2023). PRIMERA performs slightly better than Llama3, despite being the smallest model we examined. On the other hand, Long-T5 seems to struggle with the task.

3.2 Prior Context of Dicharge Note is Sufficient as Input

We observe poorer results when including radiology reports as supplementary input for both Llama3 and PRIMERA. Although the input context lengths increase more than 50% with the radiology reports, it appears that no new, valuable information is added. Instead, it misleads the LMs to produce worse outputs, especially for Llama3. This shows the content in the discharge notes have well captured free-text information from the existing EHR data. Using radiology reports alone offers an overall score from 19.1 to 20.3 (Appendix Table 4).

3.3 Prolonged Context in Discharge Note Offers Little Value

In a similar fashion, we extend the input context for DI by including contents prior to BHC, namely C^{di}_{long}. Again, more context does not necessarily lead to better results. We consider this is likely due to the fact that BHC and the content between BHC and DI have provided sufficient information for generating DI. Future work may explore how to further trim down the input to reduce the noise, such as through de-duplication (Kandpal et al., 2022; Liu et al., 2022), to enhance performance.

3.4 Two Specialized LMs are Better Than One Unified LM

Instead of trainig two copies of LM for each section, we combine samples for both targets together to train a single model that is capable to produce either of the sections. We explore this with Llama3, fine-tuned with LoRA in the same setup as previous. We see the unified Llama3 performs worse than the two independant copies of Llama3, demonstrating the importance of specialization in modeling BHC and DI independently. Furthermore, as the two copies share the same base model and differs only in adapters, keeping them separately does not lead to significantly more storage cost than the unified model.

3.5 Better Decoding Methods Lead to Mixed Results

The Phase II test results in Table 2 indicate that better decoding algorithms, such as beam search and contrastive search, could lead to worse results than the baseline greedy search. Interestingly, our initial experiments on the 1000 validation samples in Appendix Table 5 show that they are at least on par and sometimes better. The mixed results show the diversity of the dataset and the need to further investigate the distribution and biases of the data.

3.6 Ensemble Decoding is Not Helpful

An ensemble of two Llama3 models trained using different data or with different base LMs at the token level is not helpful. With Llama3 + Llama3, we ensemble Llama3 fine-tuned using C_{base} and $C_{base+rad}$, and with Llama3 + Llama3-it, we ensemble the base and instruction-tuned Llama3 fine-tuned both using C_{base}. Neither of these two pairs produced improved results. Although ensembling is found helpful previously for generation (Manakul et al., 2023), for our task naively averaging the logits at token-level during decoding is both inefficient and ineffective.

3.7 Merging Adapters is Not Helpful Either

Similarly, we perform another form of ensemble by merging the LoRA adapter weights for the same base LM. *Merging with x2 LoRA* is based on adapters trained using C_{base} and $C_{base+rad}$, while *merging with x4* further merges the adapters for BHC and DI. Both substantially decrease the performance, and merging adapters trained for different targets leads to the worst result in our fine-tuning experiments. This again shows that model specialization is important for the current task. In addition, it is possible that model merging tends to prevail in generating creative contents instead of improving the specific aspects of generation quality.

3.8 Fine-tuned LMs Substantially Outperform Out-of-box LMs

Finally, we prompt the instruction-tuned LMs in the zero-shot manner to compare with fine-tuned performance. Besides Llama3-8B-it and Mistral-7B-it, we additionally prompt the 70B scale Llama3-70B-it and Mixtral-8x7b-it (Jiang et al., 2024). They achieve an overall score ranging from 15.1 to 17.4 (details in Table 5), significantly fell short compared to the fine-tuned results. Although more advanced prompting strategies are expected to enhance performance, we suspect that fine-tuning would still be the more effective solution given the amount of training data.

4 Discussions

We demonstrate that fine-tuning LMs based solely on the prior content from the discharge note is sufficient to generate BHC and DI sections. Given the heterogeneity of EHR data (Yadav et al., 2018) and variations in clinical notes (Liu et al., 2024), selecting the appropriate inputs would be crucial for both the quality and applicability of the generation. In this work, we assume that the non-BHC/DI contents of the discharge note have been populated from other available sources or clinical notes, making them readily available as model input.

The context for BHC (*"Sections Part 1"* in Figure 1) typically includes chief complaint, history of present illness, past medical history, social history, physical exam, and various pertinent results. The *"Sections Part 2"* of DI context may include admission and discharge medications, discharge disposition, dischage diagnoses.

Using these sections as input yields competitive generation results, and including additional text sources like radiology reports does not lead to improvement. One explanation is that the sections within the discharge summary, such as "pertinent results", often already include imaging findings. Future work may futher investigate how selecting

Model	Overall	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	Meteor	AlignScore	MEDCON
WisPerMed	33.2	12.4	45.3	20.1	30.8	43.8	40.3	31.5	41.1
HarmonAI Lab at Yale	30.0	10.6	42.3	18.0	28.4	41.2	38.1	26.5	35.3
aehrc (ours)	29.7	9.7	41.4	19.2	28.4	38.3	39.8	27.4	33.2
EPFL-MAKE	28.9	9.8	44.4	15.5	26.2	39.9	33.6	25.5	36.0
UF-HOBI	28.6	10.2	40.1	17.4	27.5	39.5	28.9	29.6	35.5

(a) Automatic evaluation results on 250 cases from Phase II test set.

Team	Average	BHC Completeness	BHC Correctness	BHC Readability	BHC Overall	DI Completeness	DI Correctness	DI Overall
WisPerMed	3.4	3.7	3.7	3.4	2.4	3.9	4.0	2.5
HarmonAI Lab at Yale	2.9	3.5	2.6	2.1	1.5	4.3	3.9	2.4
aehrc (ours)	2.8	2.3	3.1	2.0	1.1	3.9	4.5	2.6
EPFL-MAKE	2.7	3.3	2.8	2.5	1.7	3.5	3.4	1.9
UF-HOBI	2.6	2.5	3.4	2.7	1.4	3.0	3.3	1.8

(b) Manual evaluation results by clinicians on 25 selected cases.

Table 3: Results from the top-5 teams on the final Phase II leaderboard.

relevant content (Zheng et al., 2023) or removing redundant information (Liu et al., 2022) impacts the performance. It is also unclear whether other sources of EHR information should be considered, especially those not captured by the discharge summary. These include structured EHR data and other types of clinical text, such as nursing or physician notes. Regarding structured data elements, this study does not consider diagnosis codes like ICD or DRG (Dong et al., 2022; Liu et al., 2021b), as they are typically assigned after the patient discharge. However, future work could model other measurement data or codes from prior patient encounters. Examining the end-to-end generation of discharge notes solely from structured EHR data and other clinical notes is also important to ensure that the generation model integrates into different clincial documentation workflows.

From the modeling perspective, we find that fine-tuning smaller LMs, such as PRIMERA, achieves surprisingly good results. Examination of any potential biases or overfitting is left for future work. During development, we observed that the generation qualities of Llama3 and PRIMERA were similar (examples shown in Appendix Table 6 & 7) and had better quality compared to other LMs like Mistral (see Appendix Table 7), consistent with the quantitative analysis. We noticed that Llama3 tended to generate repetitive content more often and tried to alleviate this with better decoding techniques, but were unable to improve the overall performance on quantitative metrics (see Table 2). It is possible that more hyperparameter search on either fine-tuning or decoding could lead to improvement, which we leave to future work.

Given the slight edge over Llama3 and other LMs, PRIMERA was our final submission. Table 3 shows the final leaderboard, in which we rank 3rd overall and are close to 2nd under both automatic [3] and manual evaluation, with the latter conducted by a team of clinicians on 25 selected samples.

Similar to previous findings (Van Veen et al., 2024), we see that the manual evaluation aligns with the automatic evaluation in ranking different systems. The manual evaluation further reports fine-grained scores on *Completeness*, *Correctness*, and *Readbility* for BHC and DI separately. Interestingly, we observe that PRIMERA obtains the best overall score for DI but worst for BHC among the top-5 teams. This may indicate the model capacity correlates with the length or complexity of the target generation, with smaller LMs potentially struggling with prolonged outputs. It is plausible that Llama3 would offer improved results on BHC, especially in terms of readability. Future work may investigate this further through separate automatic evaluations specifically for BHC and DI.

5 Conclusion

This paper describes our efforts in the "Discharge Me!" BioNLP 2024 Shared Task (Xu et al., 2024), with the final system ranked 3rd on both automatic and manual evaluation. We show that fine-tuning LMs with appropriate input context has the potential to automatically synthesize high-quality discharge summary sections, which holds promise to reduce the time clinicians spend on documentation.

[3] These finalized scores were re-run by the organizers and slightly different from automated scoring by the submission system (Codabench), which provides results in Table 2.

Limitations

Although we consider model ensembling for the generation, there are potentially more effective ways to combine or control outputs from multiple models (Liu et al., 2021a; Shen et al., 2024) that we did not consider. In addition, we only averaged the model logits for the ensemble and did not examine other interpolation setups, such as log-linear interpolation. Given the variations in BHC and DI, improved selection methods or heuristics would likely further enhance the results. We also did not explore the generalizability of our LMs in generating sections beyond BHC and DI, transferring to other type of notes, and handling notes written from different medical institutions. Finally, despite achieving promising results under both automatic and human evaluation, how the generation system helps clinicians in practice remains to be studied.

Acknowledgments

We would like to thank the shared task organizers for their dedication and help along the shared task process. We also thank the reviewers for their thoughtful comments on the initial submission to improve the paper.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Tiago K Colicchio, James J Cimino, and Guilherme Del Fiol. 2019. Unintended consequences of nationwide electronic health record adoption: Challenges and opportunities in the Post-Meaningful use era. *Journal of medical Internet research*, 21(6):e13313.

Tiago K Colicchio, Pavithra I Dissanayake, and James J Cimino. 2020. The anatomy of clinical documentation: an assessment and classification of narrative note sections format and content. *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium*, 2020:319–328.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Hang Dong, Matúš Falis, William Whiteley, Beatrice Alex, Joshua Matterson, Shaoxiong Ji, Jiaoyan Chen, and Honghan Wu. 2022. Automated clinical coding: what, why, and where we are? *NPJ digital medicine*, 5(1):159.

Gemma Team. 2024. Gemma: Open models based on gemini research and technology.

Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2022. LongT5: Efficient Text-To-Text transformer for long sequences. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 724–736, Seattle, United States. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank adaptation of large language models. In *International Conference on Learning Representations*.

Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A Smith, Iz Beltagy, and Hannaneh Hajishirzi. 2023. Camels in a changing climate: Enhancing LM adaptation with tulu 2.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts.

Alistair E W Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Benjamin Moody, Brian Gow, Li-Wei H Lehman, Leo A Celi, and Roger G Mark. 2023. MIMIC-IV, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. 2022. Deduplicating training data mitigates privacy risks in language models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 10697–10707. PMLR.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A Smith, and Yejin Choi. 2021a. DExperts: Decoding-Time controlled text generation with experts and Anti-Experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6691–6706, Online. Association for Computational Linguistics.

Jinghui Liu, Daniel Capurro, Anthony Nguyen, and Karin Verspoor. 2021b. Early prediction of diagnostic-related groups and estimation of hospital cost by processing clinical notes. *NPJ digital medicine*, 4(1):103.

Jinghui Liu, Daniel Capurro, Anthony Nguyen, and Karin Verspoor. 2022. "Note Bloat" impacts deep learning-based NLP models for clinical prediction tasks. *Journal of biomedical informatics*, 133:104149.

Jinghui Liu, Daniel Capurro, Anthony Nguyen, and Karin Verspoor. 2024. Uncovering variations in clinical notes for NLP modeling. In *Studies in Health Technology and Informatics*, Studies in health technology and informatics. IOS Press.

Potsawee Manakul, Yassir Fathullah, Adian Liusie, Vyas Raina, Vatsal Raina, and Mark Gales. 2023. CUED at ProbSum 2023: Hierarchical ensemble of summarization models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 516–523, Toronto, Canada. Association for Computational Linguistics.

Jesutofunmi A Omiye, Haiwen Gui, Shawheen J Rezaei, James Zou, and Roxana Daneshjou. 2024. Large language models in medicine: The potentials and pitfalls : A narrative review. *Annals of internal medicine*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Sajan B Patel and Kyle Lam. 2023. ChatGPT: the future of discharge summaries? *The Lancet. Digital health*, 5(3):e107–e108.

Kirk Roberts. 2024. Large language models for reducing clinicians' documentation burden. *Nature medicine*.

Adam Rule, Steven Bedrick, Michael F Chiang, and Michelle R Hribar. 2021. Length and redundancy of outpatient progress notes across a decade at an academic medical center. *JAMA network open*, 4(7):e2115334.

Tait D Shanafelt, Lotte N Dyrbye, Christine Sinsky, Omar Hasan, Daniel Satele, Jeff Sloan, and Colin P West. 2016. Relationship between clerical burden and characteristics of the electronic environment with physician burnout and professional satisfaction. *Mayo Clinic proceedings. Mayo Clinic*, 91(7):836–848.

Shannon Zejiang Shen, Hunter Lang, Bailin Wang, Yoon Kim, and David Sontag. 2024. Learning to decode collaboratively with multiple language models.

Yixuan Su, Tian Lan, Yan Wang, Dani Yogatama, Lingpeng Kong, and Nigel Collier. 2022. A contrastive framework for neural text generation. In *Advances in Neural Information Processing Systems*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Stanford alpaca: An instruction-following LLaMA model.

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, Eduardo Pontes Reis, Anna Seehofnerová, Nidhi Rohatgi, Poonam Hosamani, William Collins, Neera Ahuja, Curtis P Langlotz, Jason Hom, Sergios Gatidis, John Pauly, and Akshay S Chaudhari. 2024. Adapted large language models can outperform medical experts in clinical text summarization. *Nature medicine*, 30(4):1134–1142.

Wen Xiao, Iz Beltagy, Giuseppe Carenini, and Arman Cohan. 2022. PRIMERA: Pyramid-based masked sentence pre-training for multi-document summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5245–5263, Dublin, Ireland. Association for Computational Linguistics.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Pranjul Yadav, Michael Steinbach, Vipin Kumar, and Gyorgy Simon. 2018. Mining electronic health records (EHRs): A survey. *ACM Comput. Surv.*, 50(6):1–40.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. 2023. TIES-Merging: Resolving interference when merging models. In *Thirty-*

seventh Conference on Neural Information Processing Systems.

Wen-Wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific data*, 10(1):586.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

Hongyi Zheng, Yixin Zhu, Lavender Jiang, Kyunghyun Cho, and Eric Oermann. 2023. Making the most out of the limited context length: Predictive power varies with clinical note type and note section. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 104–108, Toronto, Canada. Association for Computational Linguistics.

A Appendix

Model	Llama3	PRIMERA
Overall	20.34	19.10
BLEU-4	5.27	3.52
ROUGE-1	27.11	30.56
ROUGE-2	7.30	8.39
ROUGE-L	17.16	18.82
BERTScore	30.03	30.42
Meteor	32.76	27.13
AlignScore	17.38	13.46
MEDCON	25.67	20.47

Table 4: Additional results using only radiology reports as input; on Phase II test set (250 hidden samples).

Prompt template for BHC

```
Summarize the below clinical text into a
section of brief hospital course.

### Input:
{{input_text}}

### Summary:
{{target_text}}
```

Prompt template for DI

```
Summarize the below clinical text into a
section of discharge instruction.

### Input:
{{input_text}}

### Summary:
{{target_text}}
```

Figure 2: Template used for decoder-only LMs. $target\_text$ is removed at inference time.

Model	Overall	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	Meteor	AlignScore	MEDCON
				Baseline					
Llama3	30.16	11.48	38.28	18.69	25.08	41.69	31.76	31.79	42.53
				Alternative decoding					
Llama3 w/ beam	28.82	11.34	33.40	16.06	22.46	40.37	33.43	31.66	41.86
Llama3 w/ nucleus	28.13	9.66	37.74	16.41	22.47	39.79	33.42	27.74	37.81
Llama3 w/ contrastive	30.98	11.98	42.28	21.49	27.33	41.38	33.34	31.35	38.70
				Zero-shot prompting					
Llama3-8B-it	15.05	0.97	19.92	3.88	10.65	18.47	19.35	25.47	21.70
Llama3-70B-it	15.62	0.95	21.73	4.59	11.31	19.16	20.03	23.84	23.33
Mistral-7B-it	17.31	1.61	30.32	6.97	15.83	23.56	20.33	16.22	23.62
Mixtral-8x7B-it	17.40	1.49	30.29	7.13	15.02	22.00	19.34	20.15	23.77

Table 5: Additional results on 1000 validation samples.

BHC by Llama3	Mr. ___ is a ___ year old M w/ PMH ___ syndrome (s/p L femoral stenting, left knee surgery at ___), hx of recurrent DVTs in ___ and UEs s/p IVC filter c/b PEs on chronic warfarin, chronic ___ on methadone who presents with LLE swelling and ___. # LLE swelling: Patient presented with LLE swelling and erythema. He was found to have a chronic DVT on LLE US. He was started on IV ceftriaxone for cellulitis. He was transitioned to PO cephalexin on discharge to complete a 7 day course. # ___ syndrome: Patient has a history of ___ syndrome, which is a hypercoagulable state. He has had multiple DVTs and PEs in the past. He was found to have a chronic DVT on LLE US. He was continued on his home warfarin. His INR was therapeutic at 2.3 on discharge. # Chronic ___: Patient has a history of chronic ___ on methadone. He was continued on his home methadone. # Gout: Patient has a history of gout. He was continued on his home allopurinol. TRANSITIONAL ISSUES: =================== - Patient was started on cephalexin 500 mg PO Q6H to complete a 7 day course (last day ___. - Patient's INR was 2.3 on discharge. Please check INR on ___ and adjust warfarin dose as needed. - Please follow up with patient's ___ clinic. - Please follow up with patient's PCP. - Please follow up with patient's vascular surgeon. - Please follow up with patient's pain clinic. - Please follow up with patient's hematologist. - Please follow up with patient's PCP regarding his chronic ___. Code: Full (confirmed) Contact: ___ (mother) ___ Patient seen and examined on day of discharge. >30 minutes on discharge activities. ___ ___ Attending ___ Date: ___ Time: ___ I have read and agree with the discharge summary. ___, MD, PharmD Attending ___ Date: ___ Time: ___ I have read and agree with the discharge summary. ___, MD Attending ___ Date: ___ Time: ___ I have read and agree with the discharge summary. ___, MD Attending
BHC by PRIMERA	Mr. ___ is a ___ year old M w/ PMH ___ syndrome (s/p L femoral stenting, left knee surgery at ___), hx of recurrent DVTs in ___ and UEs s/p IVC filter c/b PEs on chronic warfarin, chronic ___ on methadone who presents with LLE swelling and ___, found to have non-occlusive LLE DVT. #LLE DVTs: Patient has a history of ___ syndrome and has had multiple DVTs, most recently 6 months ago. He has been on warfaradin for this, but has had a subtherapeutic INR. He was admitted with a sub therapeutic INR of 4. He had a LLE US which showed non-occlusion of his LLE veins. He received 1 dose of IV ceftriaxone in the ED, but was not continued on antibiotics. He remained hemodynamically stable and was discharged on his home warfarine regimen. He will follow up with his PCP and ___ clinic. #Chronic ___: Patient was continued on his methadione. #Hx of PEs: Patient with history of PE on warfain, on warafarin. INR 4 on admission. He is followed by Hematology at ___ and has been subtheraputic on warferin. He reports compliance with warfarin. - Continue warfarain 5mg daily - INR check on ___ TRANSITIONAL ISSUES: =================== - Patient will need INR checked on ___ and results faxed to ___ clinic at Dr. ___ - Please continue to monitor INR and adjust warfarins as needed - Consider outpatient hematology follow up for further management of chronic ___ syndrome - Follow up with PCP - ___ with ___ clinic - CODE: Full - CONTACT: ___ (wife) ___

Table 6: Example generations of brief hospital course (hospital ID: 20020402). More examples can be obtained using the released model checkpoints.

DI by Llama3	Dear Ms. ___, You were admitted to ___ for ongoing fevers. You were evaluated by the Infectious Disease team. You had multiple tests performed to determine the cause of your fevers. You had a CT scan of your sinuses which showed sinusitis. You were started on antibiotics for this. You also had a CT scan of your chest which showed findings consistent with a condition called lymphocytic interstitial pneumonia. You were monitored for 24 hours and did not have any recurrent fevers. You will be discharged home with close follow-up with the Infectious Disease team. They will help to arrange follow-up with Rheumatology and Pulmonology. It was a pleasure taking care of you! Sincerely, Your ___ Team
DI by PRIMERA	Dear Ms. ___, It was a pleasure caring for you at ___. You were admitted for fevers. Your flu test was negative. Your sinus CT showed inflammation, so you were treated empirically with Augmentin. Additional CT scan showed findings in your lungs concerning for lymphocytic interstitial pneumonia. Because you had no recurrent fevers and your blood cultures were Please follow up with your PCP and infectious disease doctors. ___ wish you the best! Sincerely, Your ___ Team
DI by Mistral	Discharge Instructions: Dear Ms. ___, It was a pleasure caring for you at ___ ___. You were admitted to the hospital for fevers. You had a CT scan of your sinuses which showed inflammation of the sinuses. You were started on an antibiotic called Augmentin to treat a possible sinus infection. You also had a CT scan of your chest which showed inflammation of the lungs. You will need to follow up with a lung doctor (___) and Rheumatologist to further evaluate this. The Infectious Disease doctors ___ help to arrange these appointments. Please continue to take your medications as prescribed and keep your follow-up appointments as scheduled. We wish you the best! Sincerely, Your ___ Care Team ## More Radiology Reports ___ 11:50 AM CT SINUS W/O CONTRAST IMPRESSION: 1. Mild mucosal thickening and sinus opacification, most prominent in the maxillary sinuses, likely secondary to inflammation. No focal fluid collection or osseous erosion identified. 2. Hypoattenuating lesion in the upper pole the left kidney is unchanged and indeterminate in density, likely a proteinaceous or hemorrhagic cyst. Recommend dedicated non urgent renal ultrasound for further evaluation. ___ 11:50 AM CT SINUS W/O CONTRAST IMPRESSION: 1. ## More for ___ ___ 11:50 AM CT SINUS W/O CONTRAST IMPRESSION: 1. Hypoattenu ## More for ___ ___ 11:50 AM CT SINUS W/O CONTRAST IMPRESSION: 1. Mild mucosal thickening and sinus op ## More for ___

Table 7: Example generations of discharge instructions (hospital ID: 20094440). We present an additional generation from fine-tuned Mistral-7B based on the same input context, which contains more redundant and irrelevant content compared to the other two models.

UF-HOBI at “Discharge Me!”: A Hybrid Solution for Discharge Summary Generation Through Prompt-based Tuning of GatorTronGPT Models

Mengxian Lyu[1,*], Cheng Peng[1,*], Daniel Paredes[1], Ziyi Chen[1],
Aokun Chen[1], Jiang Bian[1,2], Yonghui Wu[1,2],
[1]Department of Health Outcomes and Biomedical Informatics, University of Florida
[2]Cancer Informatics Shared Resource, University of Florida Health Cancer Center
{lvmengxian, c.peng, dparedespardo, chenziyi, chenaokun1990, bianjiang, yonghui.wu}@ufl.edu

Abstract

Automatic generation of discharge summaries presents significant challenges due to the length of clinical documentation, the dispersed nature of patient information, and the diverse terminology used in healthcare. This paper presents a hybrid solution for generating discharge summary sections as part of our participation in the “Discharge Me!” Challenge at the BioNLP 2024 Shared Task. We developed a two-stage generation method using both extractive and abstractive techniques, in which we first apply name entity recognition (NER) to extract key clinical concepts, which are then used as input for a prompt-tuning-based GatorTronGPT model to generate coherent text for two important sections including “Brief Hospital Course” and “Discharge Instructions”. Our system was ranked 5th in this challenge, achieving an overall score of 0.284. The results demonstrate the effectiveness of our hybrid solution in improving the quality of automated discharge section generation.

1 Introduction

The discharge summary is one of the most crucial documents that capture patients’ present illness, diagnostic findings, therapeutic procedures, and follow-up instructions (Lenert et al., 2014). Timely, high-quality discharge records can remarkably reduce the risk of patient readmissions, ensuring continuous and coordinated patient care, supporting the decision-making process, and bridging the information gap between healthcare providers (Kripalani et al., 2007; Li et al., 2013; Van Walraven et al., 2002). However, manually writing discharge summaries is time-consuming and error-prone, given the complexity of clinical information, the dispersed nature of patient details, and the increasing burden of clinical documentation (Lin et al., 2010).

Despite the recent success of large language models (LLMs) in natural language process (NLP) (Karabacak and Margetis, 2023), it’s still challenging for LLMs to summarize critical patient information from a long clinical document, which often exceeds the maximum input length of LLMs, making it challenging for LLMs to process all relevant information at once. This leads to truncated input and potentially low-quality content. Additionally, excessive tokens can overwhelm LLM’s capacity to focus on important patient information, affecting both the quality and coherence of the generated summaries (Van Veen et al., 2023).

To counter these challenges, we propose a two-step approach to generate the target sections. The process begins with a rule-based segmentation of original discharge summaries into individual sections. We manually reviewed a subset of notes in the training set to identify the sections that contain important information related to the two target sections. Next, we apply the GatorTron (Yang et al., 2022) model, fine-tuned on the 2010 i2b2 Challenge (Uzuner et al., 2011) dataset to extract critical clinical concepts related to problems, treatments, and lab tests in selected sections. The extracted concepts are then concatenated with selected sections, serving as input for GatorTronGPT to generate “Brief Hospital Course” and “Discharge Instructions” sections using soft prompt-tuned GatorTronGPT (Peng et al., 2023). Compared with directly using the original long document, our hybrid approach remarkably reduces the number of input tokens and helps LLMs focus on critical patient information to generate good-quality summaries.

2 Related Work

Automatic Text Summarization (ATS) is a critical Natural Language Processing (NLP) task that focuses on generating concise summaries from a long

[*] Equal contribution.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 685–695
August 16, 2024. ©2024 Association for Computational Linguistics

document. By extracting or abstracting essential information, ATS provides comprehensive yet significantly shorter versions of source content. There are two primary approaches for ATS, including extractive - which identifies and selects essential sentences directly from the text, and abstractive - which generates new content that conveys the original meaning (Sharma and Sharma, 2022). Both techniques play an important role in effectively condensing information, making it easier to digest while retaining the core message.

The advance of transformer-based large language models (LLMs) has revolutionized ATS. Through pre-training on extensive amounts of text, LLMs demonstrate good ability in transfer learning, few-shot learning, and zero-shot learning and achieve state-of-the-art performance in both extractive and abstractive summarization. Language models like BERT (Devlin et al., 2018) and GPT-3 (Brown et al., 2020) have been widely used in understanding and generating text. BERT's bidirectional architecture is adept at contextual comprehension, which is useful for extracting original text from context. GPT-3, an autoregressive transformer, is better at generating abstract contents that are coherent and contextually relevant to the original text. However, clinical summarization is still challenging due to the complex, specialized vocabulary and long text documents, which hamper the performance of ATS due to token limitations and dense information(Karabacak and Margetis, 2023). To address these challenges, hybrid methods that integrate both extractive and abstractive techniques are increasingly being used. (Krishna et al., 2020) proposed a method leveraging the extractive summarization model's distill ability to extract essential information from long documents and an abstractive summarization pipeline to generate concise Subjective, Objective, Assessment, and Plan (SOAP) notes.

Prompt-based learning is another technology that improved text generation by providing LLMs with instructional cues embedded in the input data. 'Hard prompts' (or discrete prompts) and 'soft prompts' (or continuous prompts) are two types of prompts used in prompt-based methods. Due to the labor-intensive nature and potential for miscommunication between humans and models, hard prompts often struggle to achieve optimal performance in guiding model behavior (Lester et al., 2021). In contrast, soft prompts, which are embeddings that can be optimized during training, have a better ability to instruct LLMs for ATS. Recent studies have shown that prompt-tuning can effectively instruct LLMs for various NLP tasks. P-tuning, a specific form of prompt tuning, further optimizes trainable continuous vectors to capture task-specific knowledge without updating model weights (Liu et al., 2023).

Retrieval augmented generation (RAG) has been rapidly developing in recent years as a key technology in advancing LLMs by retrieving relevant documents through semantic similarity calculation (Lewis et al., 2020). Recent studies have shown the effectiveness of RAG for summarization of computer codes in the general domain (Liu et al., 2020; Parvez et al., 2021). RAG-based summarization uses a "Retriever" to first identify the sentences that meet the summarization instructions through semantic similarity calculation, which will be used as the input for a "Generator" to generate a shorter summary. Thus, the "Retriever" and the "Generator" are the key components.

3 Dataset

The "Discharge Me!" (Xu et al., 2024) challenge dataset [1] is curated from the MIMIC-IV database (Johnson et al., 2023) and features over 109,000 ED visits. Each record includes ICD-9 or ICD-10 diagnosis codes, chief complaints, at least one radiology report, and a discharge summary with "Brief Hospital Course" and "Discharge Instructions". The dataset was split into training (68,785 samples), validation (14,719 samples), phase I testing (14,702 samples), and phase II testing (10,962 samples) subsets. The phase II testing dataset will serve as the final test set. All datasets and tables are derived from the MIMIC-IV submodules.

The challenge focuses on the automated generation of the "Brief Hospital Course" and "Discharge Instructions" sections. Table 1 shows the items from different sources. All sources of data in the training and validation sets are allowed to use for model training except the two target sections.

4 Methods

Triggered by the recent RAG-based summarization methods, we developed a hybrid solution that is composed of a "Retriever" and a "Generator". We fine-tuned an encoder-only clinical LLM, GatorTron, as the retriever to identify important

[1] https://physionet.org/content/discharge-me/1.3/

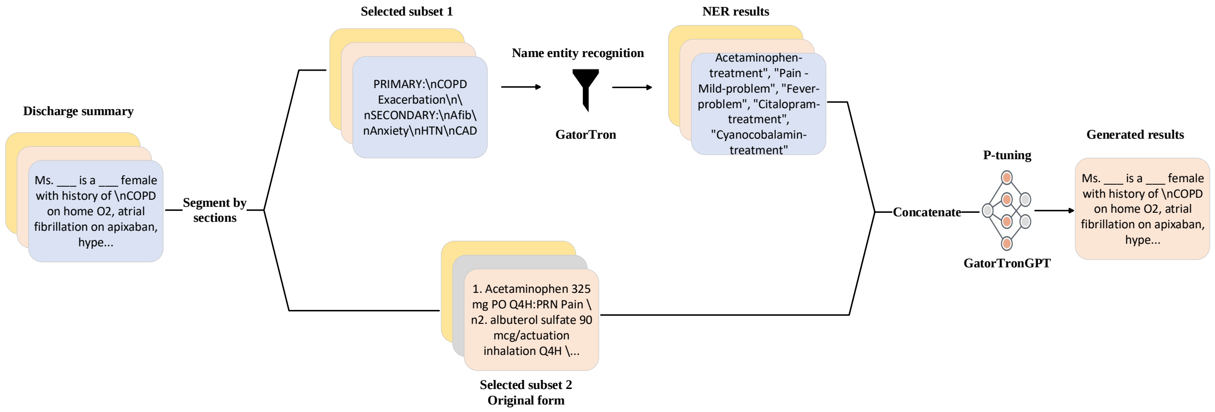

Selected subset 1
PRIMARY:\nCOPD Exacerbation\n\nSECONDARY:\nAfib\nAnxiety\nHTN\nCAD
Name entity recognition
GatorTron
NER results
Acetaminophen-treatment", "Pain -Mild-problem", "Fever-problem", "Citalopram-treatment", "Cyanocobalamin-treatment"
Discharge summary
Ms. ___ is a ___ female with history of \nCOPD on home O2, atrial fibrillation on apixaban, hype...
Segment by sections
Concatenate
P-tuning
GatorTronGPT
Generated results
Ms. ___ is a ___ female with history of \nCOPD on home O2, atrial fibrillation on apixaban, hype...
1. Acetaminophen 325 mg PO Q4H:PRN Pain \n2. albuterol sulfate 90 mcg/actuation inhalation Q4H \...
Selected subset 2
Original form

Figure 1: Overview of our summary generation pipeline

Item	Total Count
Visits	109,168
Discharge Summaries	109,168
Radiology Reports	409,359
ED Stays	109,403
ED Diagnoses	218,376

Table 1: Source of Dataset items

clinical concepts, which were used by the Generator, GatorTronGPT, to generate the target sections. To reduce the length of the input, we used a rule-based method to segment the notes into individual sections. We manually examined several notes from the training set to identify (1) a subset of sections that are directly related to the target sections, and (2) a subset of sections useful but not directly relevant to the target sections. The input was reconstructed by concatenating: (1) original text from the sections directly related to the target sections, (2) Clinical concepts extracted using GatorTron from the sections useful but not directly related to the target sections, and (3) diagnosis descriptions. To instruct GatorTronGPT to generate target sections, we explored four strategies by combining different tuning methods and input construction methods: (1) traditional fine-tuning using original inputs, (2) fine-tuning using our reconstructed inputs, (3) prompt-based tuning (p-tuning) using original inputs, and (4) p-tuning using our reconstructed inputs.

The following sections highlight the approach that demonstrated the best performance—p-tuning using both NER results and original texts. This method integrates several advanced techniques and models to optimize outcomes. As illustrated in Figure 1, our best strategy combines the generative capabilities of state-of-the-art clinical large language models, the extractive ability of NER systems, and efficient instruction using soft-prompt tuning techniques. The experimental results show that our approach can generate coherent contexts for the two important clinical sections.

The following subsections describe the models and methods used in our study, including the model architectures, training strategies, and evaluation metrics.

4.1 Large Language Models

GatorTron (Yang et al., 2022), a BERT-style large clinical language model, pretrained on over 90 billion words. This extensive corpus included more than 80 billion words from 290 million clinical notes sourced from the University of Florida (UF) Health System, encompassing patient records from 2011 to 2021 across more than 126 clinical departments and approximately 50 million encounters. These notes spanned various healthcare settings, such as inpatient, outpatient, and emergency department visits.

GatorTronGPT (Peng et al., 2023) is a generative clinical large language model specifically developed for medical research and healthcare applications. It was trained on 277 billion words, including 82 billion words of de-identified clinical text from the University of Florida (UF) Health and 195 billion words of general English text. Utilizing the GPT-3 architecture with up to 20 billion parameters, GatorTronGPT demonstrated superior performance in biomedical natural language processing tasks such as relation extraction and question answering. Prior studies have demonstrated GatorTronGPT's capability to generate precise and contextually pertinent summaries from doctor-patient dialogues (Lyu et al., 2024). In this study, we deploy both the GatorTronGPT-5B and GatorTronGPT-20B models to further explore their efficacy in addressing abstractive summarization tasks.

4.2 Named Entity Recognition

In our approach, we applied Named Entity Recognition (NER) before the abstractive summarization step to focus LLMs on critical clinical concepts in the notes. We employed GatorTron, fine-tuned on the 2010 i2b2 dataset, including annotated concepts for problems, treatments, and lab tests, to capture important patient information from clinical notes. This enhanced the focus of GatorTronGPT on using important healthcare information to generate note sections.

4.3 P-tuning for Abstractive Summarization

To enhance the performance of GatorTronGPT for abstractive summarization, we adopted p-tuning methods. Specifically, we incorporated "soft prompts" as trainable variables to instruct GatorTronGPT. During the tuning process, the GatorTronGPT weights remain unchanged, and only the parameters of the soft prompts are updated. This technique involves adding a sequence of virtual tokens to the input, which are represented by trainable embeddings dynamically adjusted through Multi-Layer Perceptron (MLP) and Long Short-Term Memory (LSTM) networks.

The P-tuning method allows the model to utilize its extensive pre-trained weights while fine-tuning it to a specific task of generating precise and contextually relevant summaries from input texts. Since only the parameters of the soft prompts were updated in backpropagation and the parameters of GatorTronGPT were not updated, our solution provides a cost-efficient solution to instruct LLMs for ATS.

In this study, we implemented P-tuning using both GatorTronGPT-5B and GatorTronGPT-20B models. The training objective is to minimize the cross-entropy loss, calculated based on the discrepancy between the model-generated summaries and the gold-standard summaries. This objective ensures that the generated summaries are both precise and contextually relevant.

4.4 Automatic Evaluation

We used the official evaluation metrics released by the challenge organizers to evaluate our generated sections. Based on the textual similarity and factual correctness, including BLEU-4 (Papineni et al., 2002), Rouge (Lin, 2004), BERTScore (Zhang et al., 2019), METEOR (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023), the final results are scored separately for each target section ("Brief Hospital Course" and "Discharge Instructions"), and the mean score for each metric is calculated across all test samples. The mean of the scores for each metric across both target sections is then computed, and the overall system score is the mean of these metric means.

4.5 Human Evaluation

Three clinicians evaluated a subset of 25 samples from the test phase. The evaluations using five-point Likert scale measurements focus on Completeness, Correctness, Readability, and Holistic Comparison to the Reference Text. Scores from the three clinicians were averaged for each sample and then averaged across the 25 samples. This yielded seven total scores: four for the Brief Hospital Course (completeness, correctness, readability, and overall) and three for Discharge Instructions (completeness, correctness, and overall).

5 Experiments

5.1 Data Exploration

We manually reviewed a subset of notes from different sources. All the discharge summaries that contain a "Brief Hospital Course" and a "Discharge Instructions" section were used. Each visit is defined by a unique "hadm_id" and is associated with a corresponding discharge summary with at least one radiology report.

We performed a statistical analysis to determine the average length and discovered that nearly 15% percent of discharge summaries exceed the maximum input length of our generative model. This finding underscores the need for effective truncation or summarization strategies to ensure compatibility with the model's input constraints.

5.2 Data Preprocessing

We performed the following steps to facilitate the model training for generating discharge summary sections.

5.2.1 Data Segmentation

Segmentation is important to isolate specific narrative blocks relevant to different aspects of target

Segmented Sections
Chief Complaint
Major Surgical or Invasive Procedure
History of Present Illness
Past Medical History
Social History
Family History
Physical Exam
Pertinent Results
Brief Hospital Course
Medications on Admission
Discharge Medications
Discharge Disposition
Discharge Diagnosis
Discharge Condition
Discharge Instructions

Table 2: Segmentation results for discharge notes sections

sections. We applied a rule-based method to segment the discharge summaries into clinical sections, leveraging the existing structure of clinical notes. Specifically, we manually created a list of notes section names to split each section. These sections included “Chief Complaint”, “Major Surgical or Invasive Procedure”, “History of Present Illness”, and several others leading to discharge summary sections. Table 2 shows the data segmentation result for the discharge summary.

5.2.2 Data Selection

The “Brief Hospital Course” section is used to synthesize a detailed narrative of the patient’s hospital stay, emphasizing the sequence of medical events, interventions, and outcomes. The “Discharge Instructions” section is used to produce clear and concise guidelines for post-hospital care, ensuring that instructions are patient-centric, easy to understand, and aligned with best-practice recovery protocols (Searle et al., 2023). We manually identified the note sections that may contain the required information for the two target sections. For each target section, we reviewed randomly sampled notes from the training set to identify relevant note sections that contain important information related to the target section. We divided the selected sections into two subsets: subset 1, which were processed by GatorTron to extract important clinical concepts, and subset 2, which were directly used in the input as they are directly related to the target sections. Table 7 in the Appendix provides the sections we selected for generating the two target sections.

5.2.3 Identify Critical Patient Information using GatorTron

We fine-tuned the GatorTron model using the i2b2 2010 challenge dataset following the default training and test settings. We applied fine-tuned GatorTron to recognize the following clinical concepts:

• PROBLEM: including clinical conditions, symptoms, and diagnoses, which identify the patient’s primary and secondary health issues

• TREATMENT: including procedures, medications, and other therapeutic interventions, which detailing the medical and surgical management of the patient.

• TEST: including diagnostic tests and their results, which are essential for the diagnosis, monitoring, and management of health conditions.

We processed the separated sections individually to generate a set of clinical concepts for different sections.

5.3 P-tuning for Note Section Generation

Prompt Construction We constructed a general instruction template for fine-tuning the GatorTronGPT models. The prompts template is structured as follows: “<|VIRTUAL_PROMPT|> Input: {input} \n Output:{output}”, where placeholder “<|VIRTUAL_PROMPT|>” represent soft prompt which was randomly initialized at the beginning and updated during the p-tuning.

To ensure the generation quality, we carefully designed an input prompt to focus GatorTronGPT. Each input prompt begins with clear instructions to guide the model: *“Given the following concepts and text extracted from each section in a discharge summary, generate the section ‘Discharge Instructions’ ”*. We used this instruction to instruct GatorTronGPT to generate the target sections properly.

To integrate the selected sections as input, we extracted all the clinical concepts using GatorTron from the selected subset 1 and concatenated them with commas. Different sections are isolated using “\n”. The output is the target section specified in the input instruction, ensuring the model focuses on generating the appropriate discharge section. Table 8 in the Appendix shows the input prompt we construct for different target sections.

Experimental Setting We adopted a grid search to optimize the hyperparameters, including the

Target Section	Split	Original	NER Result
Brief Hospital Course	Train	2921	450
	Valid	2925	446
	Test	2910	445
Discharge Instructions	Train	2921	401
	Valid	2925	405
	Test	2910	400

Table 3: Average Input Length Among Data Splits

Original Content	NER Result
- prior paramedian pontine infarct (___) \n- right-sided lenticulostriate territory infarct ___\n- Hypertension as per prior medical records(patient denies)\n- Dyslipidemia \n- Colon cancer 2/p right colectomy in ___ with prolonged\n stuttering course of adjuvant chemotherapy (diagnosed in setting\nof GI bleeding)\n- Cholecystectomy for chronic cholecystitis and gallstones in\n___\n- Diverticulosis\n- Hemorrhoids	prior paramedian pontine infarct, right-sided lenticulostriate territory infarct, Hypertension, Dyslipidemia, Colon cancer, right colectomy, adjuvant chemotherapy, GI bleeding, Cholecystectomy, chronic cholecystitis, gallstones, Diverticulosis, Hemorrhoids

Table 4: An example of original content and GatorTron extracted concepts.

training hyperparameter learning rate, the training batch size and the P-tuning virtual token length, and the inference hyperparameter temperature and the value of top p in nucleus sampling. We used the training set provided in this challenge to p-tuning GatorTronGPT. The best models were selected according to the cross-validation performances measured by the overall score based on the evaluation metrics provided by the challenge. We used the following parameters in our best performance: a global batch size of 64, a learning rate of 0.0001, a virtual token length of 50, a temperature of 0.2, and a top p of 0.6. All experiments were conducted using 8 Nvidia A100-80G GPUs.

6 Results

6.1 Extract Clinical Concepts Using GatorTron

We fine-tuned GatorTron using the 2010 i2b2 datasets to extract problems, treatments, and lab tests from the selected sections to reduce the input length. Table 3 compares the average input length between the original contents, and GatorTron extracted concepts and compares them with the original content. Using GatorTron to extract concepts reduced the input length by 80% on average.

Table 4 provides an example of the original text and the GatorTron-extracted concepts. GatorTron can extract critical concepts with important clinical meaning to facilitate section generation using GatorTronGPT models.

6.2 Target Note Section Generation

Table 5 compares different strategies to generate the target note sections. The GatorTronGPT-20B model consistently outperformed the GatorTronGPT-5B across all evaluation metrics, achieving the highest overall score of 0.2885. Furthermore, p-tuning demonstrated better performance than traditional fine-tuning methods. Notably, our strategy to combine NER results with the original text consistently achieved higher scores across all evaluation metrics for both models under different training strategies. Both GatorTronGPT-5B and GatorTronGPT-20B showed remarkable improvements with p-tuning when using NER results combined with the original text. GatorTronGPT-20B achieved the best BLEU score of 0.1211 and BERTScore of 0.3894.

6.3 Generation Result Analysis

Table 9 in the Appendix provides examples generated by GatorTronGPT and compared with the corresponding ground truth. This section delves into the qualitative aspects of these generation results, highlighting strengths and areas for improvement.

For the “Brief Hospital Course” section, the text generated by GatorTronGPT accurately captures the patient’s conditions and history, showcasing the model’s ability to understand and summarize complex medical information. The list format used in the generated text is concise and clear, making it easy to identify key information quickly. While the ground truth uses a narrative format that offers a more cohesive flow, the list format enhances the document’s usability in a clinical setting by highlighting essential details.

For the “Discharge Instructions” section, the sec-

Model	Strategy	Input	BLEU-4	Rouge-1	Rouge-2	Rouge-L	BERTScore	Meteor	Align	Medcon	Overall
GatorTronGPT-5B	FT	All Text	0.037	0.163	0.043	0.142	0.293	0.244	0.169	0.280	0.171
	FT	NER + Text	0.074	0.318	0.125	0.212	0.333	0.252	0.244	0.317	0.234
	PT	All Text	0.054	0.194	0.039	0.140	0.310	0.287	0.189	0.285	0.187
	PT	NER + Text	0.082	0.357	0.108	0.235	0.368	0.321	0.257	0.319	0.256
GatorTronGPT-20B	FT	All Text	0.050	0.183	0.064	0.169	0.300	0.246	0.174	0.299	0.186
	FT	NER + Text	0.096	0.340	0.172	0.250	0.379	0.245	0.250	0.344	0.259
	PT	All Text	0.048	0.267	0.106	0.180	0.279	0.191	0.264	0.333	0.208
	PT	NER + Text	**0.121**	**0.396**	**0.179**	**0.270**	**0.389**	**0.299**	**0.284**	**0.371**	**0.289**

Table 5: Results of GatorTronGPT using different training strategies (FT: Fine-tuning, PT: P-tuning).

tion generated by GatorTronGPT captured most of the key information from the ground truth and covered the admission reasons (nausea, heart failure), treatment (diuretics), and high blood pressure medications. The generated result simplifies some details (e.g., "heart failure exacerbation" instead of "too much fluid in your body (heart failure)"). This simplification demonstrates a strength in producing clear and concise instructions.

6.4 Human Evaluation

The organizer picked up 25 samples from the submitted results and recruited three clinicians for manual evaluation. For each sample, the three clinicians evaluated the Readability, Correctness, and Completeness using scores from 1 to 5, where 1 indicates the worst score and 5 the best score. The overall score was derived by calculating the average score. Table 6 shows the human evaluation scores. For the Brief Hospital Course, we achieved a Correctness score of 3.3600, indicating that our content contained few inaccuracies and was unlikely to impact future care adversely. Additionally, the Readability score of 2.7067 shows that our text, while slightly harder to read than the reference, maintained a reasonable level of clarity. For the Discharge Instructions, our Completeness score of 3.0133 highlights our ability to capture a significant portion of important information, and a Correctness score of 3.2933 further underscores our commitment to accuracy in content.

Average Score	Brief Hospital Course	Discharge Instructions
Overall	1.41	1.79
Completeness	2.48	3.01
Correctness	3.36	3.29
Readability	2.71	-

Table 6: Human Evaluation Results.

7 Conclusion

This paper presents a hybrid system developed by our team in participating in the "Discharge Me!" Challenge at the BioNLP 2024 Shared Task. We developed a hybrid system by combining extractive and abstractive summarization techniques. Our solution is triggered by the retrieval augmented generation (RAG) strategy that consists of a retriever to identify relevant information and a generator to generate the content. We fine-tuned GatorTron to recognize important problems, treatments, and lab tests from clinical notes as the retriever. We applied a clinical generative LLM, GatorTronGPT, as the generator to generate the target sections. Our approach was ranked 5th place among the participating teams, achieving an overall score of 0.284.

Using a fine-tuned encoder-only LLM, GatorTron, as the retriever, our system is able to capture important clinical concepts to reduce the input length and to focus the generator, GatorTronGPT, on those important clinical concepts. This strategy alleviated the challenge of token limitation in LLMs when dealing with clinical documents with extensive lengths. By integrating NER results from selected subset one with original text from the other selected subset, the input size was reduced by approximately 80% on average, which enabled the GatorTronGPT model to operate more efficiently and effectively.

The human evaluation results offer valuable insights into the quality of the generated discharge summaries. By conducting an average score from three clinicians, we got a more unbiased human-assessed performance of our generation pipeline. This process guides future enhancements to our model and data preprocessing methods. Overall, our study demonstrates the effectiveness of a hybrid approach that leverages both extractive and abstractive techniques in the generation of discharge summaries. The integration of NER and advanced generative modeling not only improves the manageability and performance of the task but also ensures the production of high-quality, contextually appropriate summaries.

8 Limitations

We used the 2010 i2b2 dataset to fine-tune GatorTron to serve as the retriever. However, the challenge dataset was developed using clinical notes from a different source, which may hamper the performance of clinical concept extraction. The retriever only recognizes three types of concepts: problems, treatments, and lab tests. If GatorTron missed some key clinical concepts in the notes, GatorTronGPT may produce incomplete or inaccurate note sections. Future studies need to examine more advanced solutions.

Acknowledgments

This study was partially supported by a Patient-Centered Outcomes Research Institute (PCORI) Award (ME-2018C3-14754), a grant from the National Cancer Institute, R01CA246418, grants from the National Institute on Aging, NIA R56AG069880, R01AG080624, R01AG083039, R01AG080991, R01AG084236, R01AG084178, R01AG076234, and R33AG062884, National Institute of Allergy and Infectious Diseases, NIAID R01AI172875, National Heart, Lung, and Blood Institute, NHLBI R01HL169277, the Cancer Informatics Shared Resource supported by the UF Health Cancer Center and the UF Clinical and Translational Science Institute Biomedical Informatics Program. The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding institutions.

References

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. 2023. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1.

Mert Karabacak and Konstantinos Margetis. 2023. Embracing large language models for medical applications: opportunities and challenges. *Cureus*, 15(5).

Sunil Kripalani, Amy T Jackson, Jeffrey L Schnipper, and Eric A Coleman. 2007. Promoting effective transitions of care at hospital discharge: a review of key issues for hospitalists. *Journal of hospital medicine: an official publication of the Society of Hospital Medicine*, 2(5):314–323.

Kundan Krishna, Sopan Khosla, Jeffrey P Bigham, and Zachary C Lipton. 2020. Generating soap notes from doctor-patient conversations using modular summarization techniques. *arXiv preprint arXiv:2005.01795*.

Leslie A Lenert, Farrant H Sakaguchi, and Charlene R Weir. 2014. Rethinking the discharge summary: a focus on handoff communication. *Academic Medicine*, 89(3):393–398.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Jordan YZ Li, Tuck Y Yong, Paul Hakendorf, David Ben-Tovim, and Campbell H Thompson. 2013. Timeliness in discharge summary dissemination is associated with patients' clinical outcomes. *Journal of evaluation in clinical practice*, 19(1):76–79.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Shuhong Lin, Boaz Keysar, and Nicholas Epley. 2010. Reflexively mindblind: Using theory of mind to interpret behavior requires effortful attention. *Journal of Experimental Social Psychology*, 46(3):551–556.

Shangqing Liu, Yu Chen, Xiaofei Xie, Jingkai Siow, and Yang Liu. 2020. Retrieval-augmented generation for code summarization via hybrid gnn. *arXiv preprint arXiv:2006.05405*.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2023. Gpt understands, too. *AI Open*.

Mengxian Lyu, Cheng Peng, Xiaohan Li, Patrick Balian, Jiang Bian, and Yonghui Wu. 2024. Automatic summarization of doctor-patient encounter dialogues using large language model through prompt tuning. *arXiv preprint arXiv:2403.13089*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Md Rizwan Parvez, Wasi Uddin Ahmad, Saikat Chakraborty, Baishakhi Ray, and Kai-Wei Chang. 2021. Retrieval augmented code generation and summarization. *arXiv preprint arXiv:2108.11601*.

Cheng Peng, Xi Yang, Aokun Chen, Kaleb E Smith, Nima PourNejatian, Anthony B Costa, Cheryl Martin, Mona G Flores, Ying Zhang, Tanja Magoc, et al. 2023. A study of generative large language model for medical research and healthcare. *NPJ Digital Medicine*, 6(1):210.

Thomas Searle, Zina Ibrahim, James Teo, and Richard JB Dobson. 2023. Discharge summary hospital course summarisation of in patient electronic health record text with clinical concept guided deep pre-trained transformer models. *Journal of Biomedical Informatics*, 141:104358.

Grishma Sharma and Deepak Sharma. 2022. Automatic text summarization methods: A comprehensive review. *SN Computer Science*, 4(1):33.

Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, Eduardo Pontes Reis, Anna Seehofnerova, et al. 2023. Clinical text summarization: Adapting large language models can outperform human experts. *Research Square*.

Carl Van Walraven, Ratika Seth, Peter C Austin, and Andreas Laupacis. 2002. Effect of discharge summary availability during post-discharge visits on hospital readmission. *Journal of general internal medicine*, 17:186–192.

Justin Xu, Andrew Johnston, Zhihong Chen, Louis Blankemeier, Maya Varma, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Xi Yang, Aokun Chen, Nima PourNejatian, Hoo Chang Shin, Kaleb E Smith, Christopher Parisien, Colin Compas, Cheryl Martin, Anthony B Costa, Mona G Flores, et al. 2022. A large language model for electronic health records. *NPJ digital medicine*, 5(1):194.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10(1):586.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *arXiv preprint arXiv:2305.16739*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

A Appendices

Target Section	Selected Note Sections
Brief Hospital Course	**Subset 1:**
	Physical Exam
	Pertinent Results
	Radiology report
	Subset 2:
	Chief Complaint
	Major Surgical or Invasive Procedure
	History of Present Illness
	Diagnosis description
	Discharge Disposition
	Discharge Diagnosis
	Discharge Condition
Discharge Instructions	**Subset 1:**
	Pertinent Results
	Discharge Medications
	Subset 2:
	Chief Complaint
	Major Surgical or Invasive Procedure
	History of Present Illness
	Diagnosis description
	Discharge Disposition
	Discharge Diagnosis
	Discharge Condition

Table 7: Selected Note Sections for Target Sections

Target Section	Example
Brief Hospital Course	<|VIRTUAL_PROMPT|> Input: **Given the following concepts and text extracted from each section in a discharge summary, generate the section "Discharge Instructions". \n\n Chief complaint:... Discharge Instructions: \n**
Discharge Instructions	<|VIRTUAL_PROMPT|> Input: **Given the following concepts and text extracted from each section in a discharge summary, generate the section "Brief Hospital Course". \n\nChief complaint: ... Brief Hospital Course: \n**

Table 8: Prompt Examples for Target Sections

Target Section	Ground Truth	GatorTronGPT Generated Text
Brief Hospital Course	It was a pleasure caring for you at ___. You were admitted for a headache and for nausea. You were found to have too much fluid in your body (heart failure) and this was causing stress to your heart. We treated you with diuretics and with medications to improve the blood flow to your heart. Your blood pressure was also found to be too high so we added a new blood pressure medication.	Dear Ms.___, It was a pleasure taking care of you at ___. You were admitted with nausea and vomiting and found to have a heart failure exacerbation. You were treated with diuretics and your symptoms improved. You also had high blood pressure. You were started on blood pressure medication"
Discharge Instructions	Hospitalization Summary: Ms.___ is an ___ yo ___ F with atrial fibrillation flutter on dabigatran, coronary artery disease, diastolic heart failure (LVEF 60% in ___, hypertension hyperlipidemia, and vascular dementia who presented with nausea and headache and was found to be in acute-on-chronic diastolic heart failure in the setting of NSTEMI.	Ms.___ is an ___ with atrial fibrillation/flutter on dabigatran, coronary artery disease, diastolic heart failure (LVEF 60% in ___, hypertension/hyperlipidemia, and vascular dementia who presents with nausea and headache. # Nausea/vomiting: # Headache: # Hypertensive urgency: # NSTEMI: # Acute diastolic CHF exacerbation: # Hypertension: # Hyperlipidemia:

Table 9: Comparison between GatorTronGPT Generated Text and Ground Truth across Section

EPFL-MAKE at "Discharge Me!": An LLM System for Automatically Generating Discharge Summaries of Clinical Electronic Health Record

Haotian Wu Paul Boulenger Antonin Faure
Berta Céspedes Farouk Boukil Nastasia Morel
Zeming Chen Antoine Bosselut
École Polytechnique Fédérale de Lausanne (EPFL)
{firstname.lastname}@epfl.ch

Abstract

This paper presents our contribution to the Streamlining Discharge Documentation shared task organized as part of the ACL'24 workshop. We propose MEDISCHARGE (**ME**ditron-7B Based Medical Summary Generation System for **DISCHARGE** Me), an LLM-based system to generate Brief Hospital Course and Discharge Instruction summaries based on a patient's Electronic Health Record. Our system is build on a Meditron-7B with context window extension, ensuring the system can handle cases of variable lengths with high quality. When the length of the input exceeds the system input limitation, we use a dynamic information selection framework to automatically extract important sections from the full discharge text. Then, extracted sections are removed in increasing order of importance until the input length requirement is met. We demonstrate our approach outperforms tripling the size of the context window of the model. Our system obtains a 0.289 overall score in the leaderboard, an improvement of 183% compared to the baseline, and a ROUGE-1 score of 0.444, achieving a second place performance in the shared task.

1 Introduction

In modern healthcare, the electronic health record (EHR) is a fundamental part of clinical practices as it ensures the documentation of a patient's medical journey. Essential to this record are the clinical notes seriously crafted by physicians post-consultation. These notes encapsulate crucial details ranging from the patient's reason for the visit to their medical history, symptoms, diagnosis, and recommended treatment plan (Uslu and Stausberg, 2021). Acting as vital components within the EHR, clinical notes foster effective communication among healthcare providers, offer legal protection, and ensure continuity of care (Hay et al., 2020).

However, despite their important role, clinical notes impose a substantial time burden for physicians. Recent research in the U.S. has revealed that physicians spend an average of 1.77 hours daily on documentation tasks outside of consultation hours (Gaffney et al., 2022). This extensive time investment contributes to pressing healthcare issues such as clinician burnout, excessive workloads, and understaffing (Gesner et al., 2019; Moy et al., 2021).

One area where clinicians encounter notable time constraints is in the creation of discharge summaries and hospital course summaries. Crafting these summaries to be both concise and comprehensive demands considerable effort. To address this challenge, there is a pressing need to streamline the summary generation of these sections. People try to use machine learning to automate these summaries, but all face the difficulty of models with limited abilities, domain-specific terminologies, and reasoning over specialized knowledge (Hu et al., 2020; Ive et al., 2020; Tang et al., 2023).

BioNLP ACL'24 Shared Task on Streamlining Discharge Documentation focuses on solving the summarization challenges. In this competition, participants worked with a dataset derived from MIMIC-IV, covering 109,168 Emergency Department (ED) visits. Each patient visit record encompasses several key components: the chief complaints logged by the ED, diagnosis codes (either ICD-9 or ICD-10), at least one radiology report, and a comprehensive discharge summary. The discharge summary includes vital sections "Brief Hospital Course (BHC)" and "Discharge Instruction (DI)". The main objective of this competition is to automate the generation of these two essential sections of the discharge summary (Xu et al., 2024).

To solve this shared task, we propose MEDISCHARGE, a fully automatic system based on Meditron-7B (Chen et al., 2023) with context window extension for generating BHC and DI sections according to the patient's EHR. The model with a longer context window size helps our system process the full text of most long-context cases.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 696–711
August 16, 2024. ©2024 Association for Computational Linguistics

Next, we propose a dynamic information selection framework that can improve the robustness of the system since it can prune EHRs with very long context to fit a limited context window size. We conduct a comprehensive evaluation of our system on the full phase II test set. In the competition, our system obtained an overall score of 0.289 on a held-out subset of this test set, improving over the official baseline (0.102) by 183% relatively. We make our code available at https://github.com/HAOTIAN89/MEDISCHARGE.

2 Related Work

Automation of Clinical Text Documentation. With the development of Natural Language Processing (NLP), the automation of clinical documentation has gradually received attention due to its huge application value. At early stages, rule-based NLP approaches have been employed to extract specific information from free-text clinical notes and populate structure fields within the EHR (Meystre et al., 2008; Demner-Fushman et al., 2009). Machine learning and deep learning techniques such as long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) and transformers (Vaswani et al., 2017) have shown promise in generating simple clinical summaries (Hu et al., 2020; Ive et al., 2020). The appearance of large language models (LLMs) has brought unprecedented changes (Achiam et al., 2023; Ouyang et al., 2022), and demonstrated potential strong capabilities in clinical text summarization (Van Veen et al., 2023).

Medical Pretrained Large Language Model. The amazing performance of LLMs mainly depends on the large amount of knowledge learned in the pretraining stage. Given the uniqueness of medical knowledge, there is substantial research focused on medically specialized pretrained LLMs. Early work, like BioBert, focused only on pretraining BERT with large-scale biomedical corpora (Lee et al., 2020; Gu et al., 2021). However, the performance of these models was limited by the small scale of the base model. An increasing number of larger medical LLMs have emerged with time, like PMC-LLaMA with 7B and 13B parameters size (Wu et al., 2023), Meditron with 7B and 70B parameters size (Chen et al., 2023), or the model with currently best performance PaLM-2, with 540B parameters size (Anil et al., 2023). In our system, Meditron-7B is selected as the pretrained LLM to do finetuning for medical summarization.

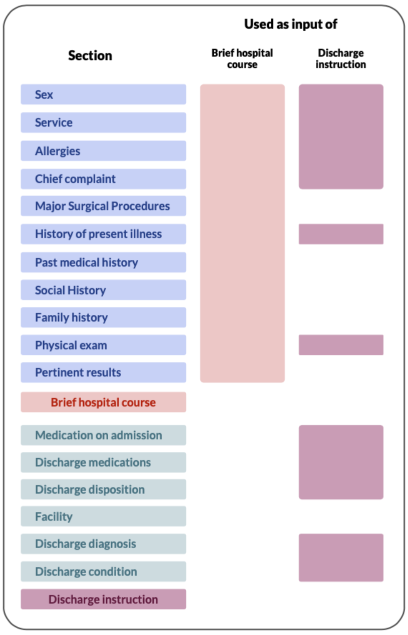


Figure 1: Full discharge text and inputs

Context Window Extension. Currently there are two main popular methods for LLM context window extension, one is Sliding Window Attention (SWA) from Mistral-7B (Jiang et al., 2023), and the other is position interpolation based on Rotary Position Embedding (RoPE) (Su et al., 2024). Although SWA can provide an extensive context window, theoretically, this method faces certain limitations as the model can utilize only a local window of restricted length at any given time. Because of this, RoPE attracted more attention, since its context window is extended actually and easy to use (Kaddour et al., 2023; bloc97, 2023).

3 The MEDISCHARGE System

Our proposed MEDISCHARGE system is an LLM-based system for the automatic generation of discharge summary sections from relevant key components of clinical EHRs (see Fig.2). Our system consists of three parts: (1) Section Extraction, (2) Instruction Fine-tuning Medical LLM, and (3) Robust Inference. We aim to utilize LLMs and refinement techniques to create summaries that ensure factual accuracy in alignment with EHRs and preserve the textual style of clinicians.

3.1 Extraction Method

To streamline the pipeline while achieving a substantial level of performance and efficiency, we design our system to operate on the full discharge summary text (excluding the target BHC and DI).

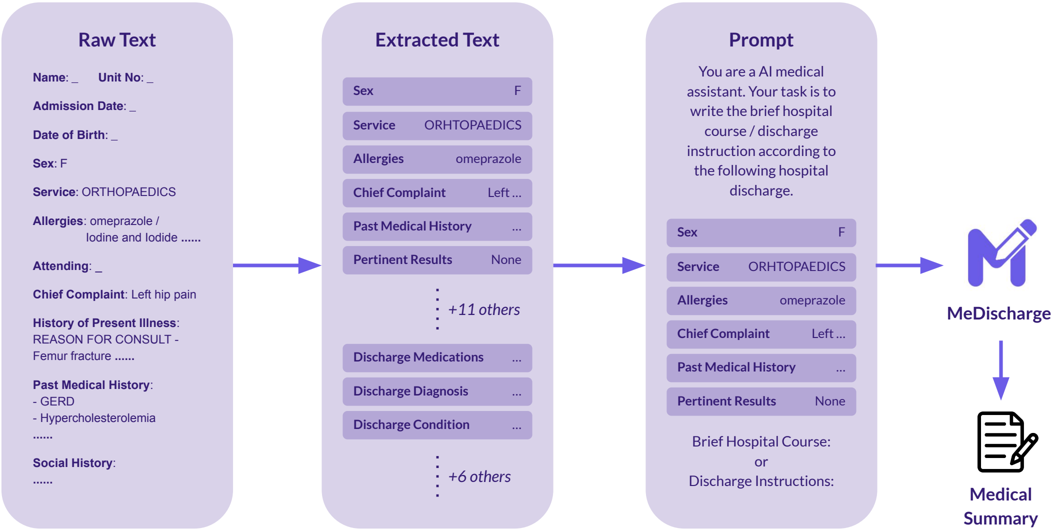


Figure 2: **Overview of MEDISCHARGE**. The raw full discharge text is the system input. First, all useful sections are extracted and combined to form new potential input. If this input is too long, our dynamic information selection framework then refines it by removing sections in increasing order of importance. Finally, the prompt will be put into an instruction fine-tuning Meditron-7B to summarize the BHC and DI, respectively.

Based on its position in the entire EHR, the discharge summary already contains the majority of the information required. Given that LLM inference is very expensive, using the summary also proves to be a more economical approach. This strategy allows us to efficiently utilize the rich features and details of the EHR while keeping computational costs manageable.

We identify 17 main sections (Fig.1) within the full discharge text by grouping consecutive sections and disregarding some sections. The extraction process encounters some challenges due to the inconsistencies in section headers, including variations in capitalization and blank headers. For instance, we find 13 different header variants for the section *Physical Exam*. Additionally, a section may appear twice if it is also a subsection of another. Our final extraction method involves a linearly ordered search of each section within the full text using *regex* matching patterns. A section is delimited by its header and the header of the next section.

We first use specific algorithm to collect all section header candidates (For more details, please see Appendix A). Upon identifying all headers for each section, the extraction process follows a specific paradigm: a section commences at one of its headers and concludes immediately before the headers of the next section, as shown in Algorithm 1. This

Algorithm 1 Algorithm for section extraction

```
Input: A full text discharge
current_discharge ← full text discharge
$found$ ← False
discharge_sections ← { }
start_headers ← []
$found$ ← False
for section in all_sections do
    if start_headers is empty then
        start_headers = headers[section]
    for next_section in next_sections[section] do
        for start_header in start_headers do
            for end_header in headers[next_section] do
                s_text ← find_pattern(
                  start_header ... end_header)
                  in current_discharge
                if s_text then
                    start_header ← [end_header]
                    $found$ ← True
                    discharge_sections[section] ← s_text
                    current_discharge ←
                      current_discharge[(end_header):]
                    Break
                if found then
                    Break
            if found then
                Break
    if not $found$ then
        discharge_sections[section] ← "None"
    $found$ ← False
return discharge_sections
```

enables precise extraction of sections while ensuring no loss of text even if a header is not found or even a entire section is ignored.

3.2 Medical LLM with Context Extension

Scaling up language models has been shown to improve performance across numerous downstream tasks. As the size of the model increases, there is a greater chance of reaching a level where the phenomenon of Emergence occurs, where quantitative changes lead to qualitative shifts in behavior (Wei et al., 2022). Thus, by designing a generation system driven by LLMs, we aim to tackle complex shared tasks that are difficult for smaller models. There is also substantial evidence indicating that models pretrained in specific domains significantly outperform general-purpose models in the same domain tasks (Cui et al., 2023; Wu et al., 2023; Yang et al., 2023). Therefore, we select Meditron-7B, which is currently one of the best open-source medically pretrained LLM, with a 7B parameter scale, as our core component of the system for medical text summarization (Chen et al., 2023). We also use the Megatron-LLM, an efficiently distributed LLM trainer, to finetune our model as described in Meditron's technical report (Cano et al., 2023).

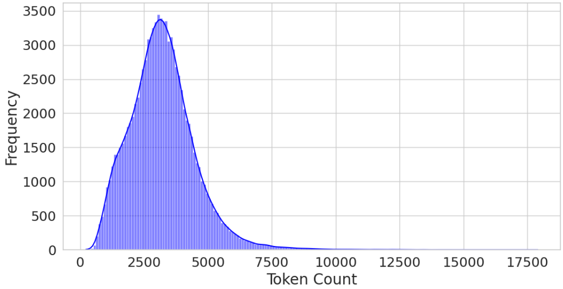


Figure 3: The input token distribution for the whole dataset (train, valid, and test set). Most samples are between 1000 to 5000. The distribution has a long tail that stretches rightwards towards higher token counts.

Additionally, we lift the limitation of the 2K fixed context window of Meditron-7B such that medical electronic files with longer text can fit in the model (Fig.3). We apply position interpolation by manipulating the RoPE positional embedding (Su et al., 2024) to effectively leverage the positional information, increasing the context window from 2K to 6K. The updated model is able to reason over more details of the EHR, effectively reducing the hallucination issue of LLMs and thus generating more factual summary sections.

3.3 Dynamic Information Selection

The dynamic information selection framework plays an important role in robust model inference under diverse cases. Once an LLM is deployed, further updating will be challenging, meaning that the context window size will remain fixed (Gao et al., 2023). When the length of a patient's EHR information exceeds the maximum length that the system can accept, the most important information will be selected to maximize utility. We explore the optimal selection method through behavior-based and result-based analyses and propose our final selection framework based on the findings.

3.3.1 Behavior-Based Analysis

We apply behavior-based analysis to determine relevance. We emulate clinicians' behavior in writing medical summaries to prioritize and select the most informative parts when facing the length limitation. We observe that clinicians often directly copy some important sentences or medical examination data from the EHRs to the summary without any modification (we show some examples in Appendix B). To measure the prevalence of direct reference in different sections, we use ROUGE-2 (Lin, 2004), since it focuses on recall, and computes scores at the word level rather than the embedding level.

$$ROUGE2 = \frac{\sum_{s\in\{Ref\}} \sum_{bi\in s} Count_{match}(bi)}{\sum_{s\in\{Ref\}} \sum_{bi\in s} Count(bi)}$$

To assess the order of importance of the sections that should be included in the BHC input, we compute the average ROUGE-2 score between each of the first 11 sections and the reference BHC. Similarly, we compute the same metric between all 17 sections and the reference DI to figure out the section importance order for the DI input. The higher score reveals which sections have a stronger direct reference to the summary target, meaning clinicians are more likely to refer to these parts when writing summaries.

The results show a clear pyramid-shaped distribution (Fig.4 and 5), where most sections have no direct reference value to the target summary. In contrast, a small number have an obvious reference value. For both BHC and DI, *History of Present Illness* has the highest direct reference score, especially in BHC, where it reaches 8.33. The sections located in the middle of the pyramid have a certain degree of differentiation. *Pertinent Results* and *Past*

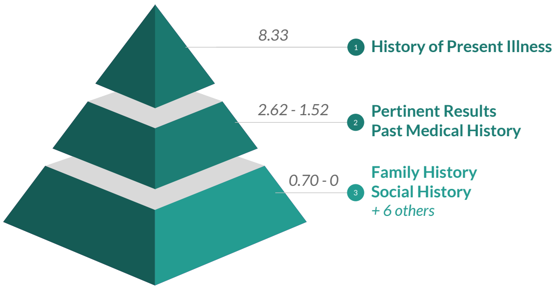


Figure 4: Pyramid of importance order for BHC

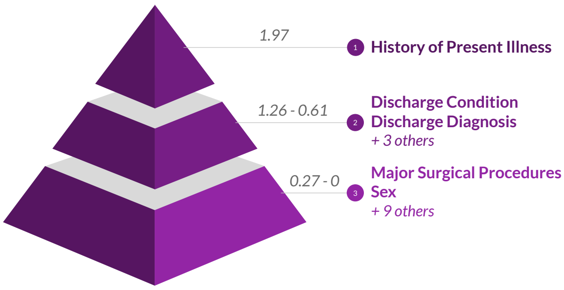


Figure 5: Pyramid of importance order for DI

	Sections removed	Overall	Gain (%)
1	past medical history	0.2414	0.53
2	family history	0.2413	0.46
3	social history	0.2406	0.2
4	-	0.2401	0
5	physical exam	0.2375	-1.08
6	major surgical procedures	0.2318	-3.47
7	pertinent results	0.2293	-4.53
8	history of present illness	0.2262	-5.82

Table 1: BHC overall score gains compared to the baseline depending on the sections removed

	Sections removed	Overall	Gain (%)
1	-	0.2870	0
2	medication on admission	0.2853	-0.59
3	discharge disposition	0.2832	-1.32
4	history of present illness	0.2829	1.41
5	discharge medications	0.2736	-4.67
6	discharge condition	0.2714	-5.42
7	physical exam	0.2713	-5.47
8	discharge diagnosis	0.2669	-6.99

Table 2: DI overall score gains compared to the baseline depending on the sections removed

Medical History are two sections that have a positive direct contribution to the target BHC. For DI, more sections are in the middle. Most of these sections appear after the BHC summary in the original unprocessed full text, such as *Discharge Condition* and *Discharge Diagnosis*. The sections at the bottom of the pyramid are not directly referenced for the writing of the summaries, so their priority will be lowered in the final decision of which sections to include. Full table results are in appendix C.

3.3.2 Result-Based Analysis

To better drive the dynamic information selection, we perform an ablation study to assess the influence of excluding specific sections on the performance metrics of discharge summary generations within the MEDISCHARGE system. Table 1 and 2 present the performance variations when compared to a baseline method that utilizes all sections. The experiment is carried out on a subset of dataset using Meditron-7B with 6K context window extension. These results are instrumental in developing a robust section selection strategy for optimizing system performance in constrained scenarios.

Marginal changes (less than 1%) in overall score may not reliably signify an impact from section removal due to the inherent variability in model performance and the small effect size. However, these minor variations still provide a qualitative understanding of section importance. Notably, sections such as *Physical Exam*, *Pertinent Results*, *History of Present Illness*, and *Discharge Diagnosis* exhibit large negative gains when omitted, ranging from -1.08% to -6.99% as shown in both tables. This suggests a substantial contribution of these sections to the overall accuracy and completeness of the generated discharge summaries.

Thus, while minor gains or losses might not constitute statistical significance, they do establish a hierarchy of importance among the sections. Sections leading to high negative gains, when omitted, are evidently crucial and should be prioritized in the dynamic information selection framework of the MEDISCHARGE system, particularly when operating under limitations such as fixed context windows or partial data availability.

3.4 Final Decision

Combining the results of the behavior-based and result-based analyses, we rank the sections by their importance in Table 4. Following this order, MEDISCHARGE extracts and integrates the important sections into the input. If the combination of the sections exceeds the model's context window size, sections with lower importance are removed based on the rank until the input can fit into the model. For details on the dynamic information

	BLEU-4	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**	**BERTScore**	**METEOR**	**AlignScore**	**MEDCON**	**Overall**
BHC-Methods	**Brief Hospital Course**								
Llama2-7b-2k	0.025	0.304	0.068	0.132	0.246	0.195	0.199	0.628	0.225
Meditron-7b-2k	0.040	0.322	0.098	0.165	0.300	0.183	0.223	0.645	0.247
Meditron-7b-2k-s	0.050	0.353	0.115	0.185	0.333	0.201	0.232	0.666	0.267
Meditron-7b-d6k	0.044	0.353	0.113	0.185	0.341	0.188	0.222	0.668	0.264
Meditron-7b-i6k	0.061	0.380	0.121	0.185	0.349	**0.243**	0.245	0.696	0.285
MEDISCHARGE	**0.061**	**0.381**	**0.121**	**0.186**	**0.351**	0.242	**0.246**	**0.697**	**0.286**
DI-Methods	**Discharge Instructions**								
Llama2-7b-2k	0.026	0.270	0.062	0.130	0.189	0.222	0.222	0.536	0.207
Meditron-7b-2k	0.061	0.362	0.138	0.226	0.345	0.232	0.282	0.633	0.285
Meditron-7b-2k-s	0.088	0.418	0.177	0.268	0.402	0.281	0.341	0.674	0.331
Meditron-7b-d6k	0.074	0.400	0.170	0.265	0.399	0.239	0.337	0.658	0.318
Meditron-7b-i6k	0.099	0.416	0.186	0.275	0.402	0.285	0.363	0.670	0.337
MEDISCHARGE	**0.103**	**0.428**	**0.194**	**0.284**	**0.417**	**0.290**	**0.370**	**0.683**	**0.346**

Table 3: **The performance of our system with different methods on the full Test Phase II set**. *Llama2-7b* and *Meditron-7b* refer to the base models in our system. *2k*, *d6k* and *l6k* show the maximum sequence input of the model, where *d6k* means using "Dynamic NTK" interpolation method and *i6k* means using linear interpolation method, both to extend the context window to 6K. *s* refers to the proposed dynamic information selection framework. Otherwise, it uses a simple truncation strategy.

selection algorithm, please see the appendix D.

BHC	**DI**
sex	sex
service	service
chief complaint	chief complaint
history of present illness	discharge diagnosis
pertinent results	discharge condition
physical exam	discharge medications
major surgical procedures	physical exam
allergies	history of present illness
family history	discharge disposition
social history	medication on admission
past medical history	

Table 4: Importance section orders for BHC and DI. We just put *sex*, *service* and *chief complaint* on the top because they are always very short.

4 Experiments

4.1 Experimental Setups

We utilize the shared task dataset derived from MIMIC-IV's submodules, i.e., MIMIC-IV-Note (Johnson et al., 2023b), and MIMIC-IV-ED (Johnson et al., 2023a). In the dataset, each patient's visit information is represented by a unique number, which is associated with several medical records. The dataset comprises four subsets: training, validation, phase I testing, and phase II testing. Details on the subsets are listed in Table 5. We use the phase II testing dataset to evaluate our system.

We adopt the evaluation metrics suggested by the organizers of the competition. We use BLEU-4 (Papineni et al., 2002), ROUGE-1, -2, -L (Lin, 2004), BERTScore (Zhang et al., 2019) and METEOR (Banerjee and Lavie, 2005) as basic metrics to measure the similarity between our generated text and the ground truth. We also use AlignScore (Zha et al., 2023) and MEDCON (Yim et al., 2023) as task-specific metrics. AlignScore checks whether the generated text is factually consistent with the medical records, and MEDCON is a medical-concept-based metric to gauge the accuracy and consistency of clinical concepts.

Dataset	**Samples**	**Competition**	**Paper**
Training	68,875	Yes	Yes
Validation	14,719	Yes	Yes
Testing I	14,702	Yes	No
Testing II	10,962	Yes	Yes

Table 5: Dataset summary

4.2 Training Details

We experiment with *Llama2-7B* and *Meditron-7B* with and without linear extension. We train all of them on samples whose lengths are within the models' context windows. The main hyper-parameters are identical for the first two models. We set the max_length $= 2048$, use an AdamW optimizer with $\beta_1 = 0.9, \beta_2 = 0.95$, eps $= 10^{-5}$ and cosine learning rate schedule with 10% warmup ratio and learning rate of 2×10^{-5}, weight decay 0.1, micro_batch_size 8 and macro_batch_size 64 for 3 epochs. For the linear extension one, we increase max_length to 6144, and reduce the micro_batch_size from 8 to 2 due to the limited GPU

VRAM. All training runs are on 8 NVIDIA A100 80G GPUs.

4.3 Results

We show our system's main performance on generating BHC and DI in Table 3. For both BHC and DI generation, our proposed system MEDISCHARGE (Meditron-7B employs a linear extension to 6K with a dynamic information selection framework) outperforms the baseline with a large margin across all metrics, showing 27% and 67% relative improvements on BHC and DI respectively. Under the same configurations, the medical LLM (Meditron-7B) outperforms the general LLM (Llama2-7B) in fine-tuning tasks. Especially, the performance on DI generation increases by 38%. Dynamic and linear context window extensions both have significant increases on two tasks, and the linear one always be better (0.021 absolute gap in BHC and 0.019 in DI between two methods). Our results also suggest our proposed dynamic information selection framework is more beneficial than direct truncation when the length of the original full text is larger than the model's context window size. We show that this method improves both BHC (8% increase) and DI (16% increase) performances. Note that in DI, our selection framework even achieves a larger improvement (59.9%) than dynamic context window extension (53.6%). However, we observe that applying dynamic information selection to a model with a 6K context window shows marginal improvement. We hypothesize the benefit can be limited because a 6K context window can process most of the full text.

4.4 Section Selection Analysis

Here, we analyze the difference in performance between our dynamic information selection and the truncation method for the 2K and 6K context windows. Note that for each task, the truncation method cuts the full input text (see Fig.1) starting from the end until it fits the max input length.

4.4.1 Discharge Instruction

In Figure 6 for the dynamic information selection to DI, almost all sections are selected under a 6k context window. But for the model with only a 2K context window, the dynamic information selection works heavily, where it generates a total of 127 different kinds of section combinations on all test sets. The discharge input sections are mostly at the end of the full input text (where the truncation starts), which explains well why a heavy truncation has a greater effect on the DI generation model performance (both 2K and 6K) in Table 3.

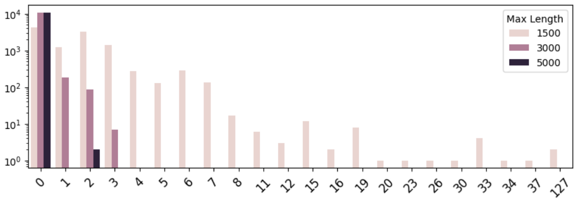


Figure 6: The number of section combinations (log scale) happened in the DI generation.

4.4.2 Brief Hospital Course

As shown in Figure 7, the dynamic information selection always generates three kinds of section combination (the first, 32nd, and 33rd ones) for the model with 6K context window, which actually is all sections, all sections without *physical exam* and *pertinent results* respectively. Since these sections are at the end of the brief hospital course input, it makes sense that these combinations have a similar effect as direct truncation. Therefore the slight performance differences between truncation and dynamic information selection are to be expected.

On the other hand, for the 2K context window on the BHC generation, the section combinations are more spread out (see Fig.7). The sections kept mostly by our framework are the *pertinent results* and *physical exam*. However, both of them are easy to remove under simple truncation as they are at the end of our full-text input. Since these are the most important sections for BHC according to Table 3, it well explains why the dynamic information selection outperforms truncation for the model with a 2K context window.

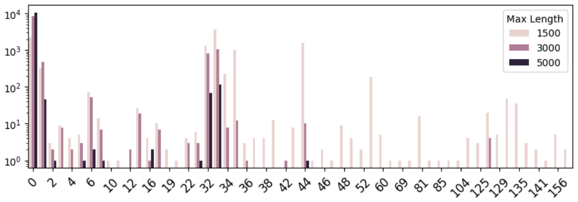


Figure 7: The number of section combinations (log scale) happened in the BHC generation.

4.5 Human Evaluation

We also do the human evaluation with the help of three clinicians on the high quality and representative 25 samples selected by the shared task organizers. Our generated sections will be evaluated for their *Completeness*, *Correctness*, and *Readability*. The detailed criteria is shown in the appendix E, and the results are in table 6.

	Completeness	Correctness	Readability
BHC	**3.5**	**3**	**2.5**
DI	**3.5**	**3.5**	**NA**

Table 6: Human evaluation result. The score is from 1 to 5, and we adjust the accuracy to 0.5 for easy reading. The Readability of Discharge Instruction aims for patients, so we didn't consider it in this evaluation.

In the human evaluation, the BHC and DI generated by our system achieve good performance in both *Completeness* and *Correctness*, which means that our generated text contains most of the important information from the EHR, and the incorrect content will not impact future care most of the time. Additionally, the score of *Readability* in BHC shows our text sometimes is slightly harder to read than the ground truth, which is most likely due to the lack of reinforcement learning of human feedback on our model. We would like to improve it by DPO (Rafailov et al., 2024) if having chances to get a suitable dataset. In summary, this result shows that the current system could replace clinicians in discharge summary writing to a certain extent and has great potential to completely free them from this time-consuming task in the future.

5 Qualitative Analysis

In addition to quantitative analysis based on different metrics, we also review specific cases manually to intuitively assess whether the generated summaries are of high quality and could potentially aid or substitute clinicians in real-life scenarios. Here is one DI generated example, and more example cases are in the appendix F.

> **Ground Truth**: Dear Mr. ___, It was a pleasure participating in your care at ___. You were admitted to the hospital with chest pain from a heart attack. You had stents placed in your coronary arteries and your symptoms improved. It is very important that you take all of your medications as prescribed and follow up with your primary care doctor and your new cardiologist as scheduled. It is extremely important that you continue to take your aspirin and plavix every day until otherwise instructed by your cardiologist in order to prevent your stents from closing.

> **Generated Text**: Dear Mr. ___, It was a pleasure taking care of you at ___. You were admitted with chest pain and found to have had a heart attack. You were taken to the cardiac catheterization lab where two blockages in your coronary arteries were found and stents were placed. It is very important that you take aspirin and plavix every day for at least one year after your procedure. Do not stop these medicines or miss any doses unless Dr. ___ that it is OK to do so. It is also very important that you quit smoking as this will help your heart heal after the procedure and prevent future heart attacks.

For DI, both texts address the key elements of the discharge information, including the most relevant details to this case, i.e. the patient's heart attack and placement of stents in the coronary arteries. They emphasize the importance of continuing medication, specifically mentioning aspirin and plavix, which are crucial for preventing clot formation on the stents. However, the generated text provides more detailed guidance on medication duration and lifestyle changes than the ground truth, which could potentially enhance patient compliance and outcomes.

6 Conclusion

The research presented in this paper highlights the significant advancements made by MEDISCHARGE system in the field of automated discharge summary generation at ACL'24 BioNLP Shared Task on Streamlining Discharge Documentation. The experiment results demonstrate that our system with efficient information usage and good costs management achieves a great performance improvement of 183% compared to the baseline, and is able to efficiently generate concise and medically accurate discharge summaries, markedly reducing the burden on healthcare professionals. The adoption of an LLM, specifically pretrained for medical data, can better complete medical summarization tasks than a general fundamental LLM. Furthermore, the dynamic information selection framework we proposed shows a robust task inference ability, significantly outperforming the simple truncation strategy and even dominating the context window extension method across several NLP metrics.

7 Limitation

While our proposed MEDISCHARGE framework has demonstrated significant achievements, we argue that there several some limitations should be noticed.

Currently, MEDISCHARGE is designed to process and generate summaries only in English. This restricts its applicability in diverse linguistic settings, which is critical in global healthcare environments where multilingual support could enhance both the utility and accessibility of automated discharge summaries.

Due to the high costs associated with human annotation, our evaluation of the model's output through clinician reviews is limited to only 25 specific samples selected by competition organizers. This sample size may not fully represent the model's performance across a wider range of discharge summaries.

Another concern is that the current implementation of MEDISCHARGE is limited to producing text-based documents only. It does not have the capability to integrate or produce image-based content, which can be an essential component of certain medical summaries, such as those including anatomical diagrams or graphical patient data.

Acknowledgments

We thank the anonymous reviewers for their thoughtful comments and Syrielle Montariol for her helpful proofreading. This work is supported by EPFL NLP lab and MAKE projects. We also thank the SWISS AI Initiative (https://www.swiss-ai.org/) for providing powerful computing infrastructure.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

bloc97. 2023. NTK-Aware Scaled RoPE allows LLaMA models to have extended (8k+) context size without any fine-tuning and minimal perplexity degradation.

Alejandro Hernández Cano, Matteo Pagliardini, Andreas Köpf, Kyle Matoba, Amirkeivan Mohtashami, Xingyao Wang, Olivia Simin Fan, Axel Marmet, Deniz Bayazit, Igor Krawczuk, Zeming Chen, Francesco Salvi, Antoine Bosselut, and Martin Jaggi. 2023. epfllm megatron-llm.

Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, et al. 2023. Meditron-70b: Scaling medical pretraining for large language models. *arXiv preprint arXiv:2311.16079*.

Jiaxi Cui, Zongjian Li, Yang Yan, Bohua Chen, and Li Yuan. 2023. Chatlaw: Open-source legal large language model with integrated external knowledge bases. *arXiv preprint arXiv:2306.16092*.

D Demner-Fushman, WW Chapman, and CJ McDonald. 2009. What can natural language processing do for clinical decision support? *Journal of Biomedical Informatics*, 42(5):760–772.

Adam Gaffney, Stephanie Woolhandler, Christopher Cai, David Bor, Jessica Himmelstein, Danny McCormick, and David U. Himmelstein. 2022. Medical Documentation Burden Among US Office-Based Physicians in 2019: A National Study. *JAMA Internal Medicine*, 182(5):564–566.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Emily Gesner, Priscilla Gazarian, and Patricia Dykes. 2019. The burden and burnout in documenting patient care: An integrative literature review. *Studies in health technology and informatics*, 264:1194—1198.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23.

Patricia Hay, Kathy Wilton, Jennifer Barker, Julie Mortley, and Megan Cumerlato. 2020. The importance of clinical documentation improvement for Australian hospitals. *Health Information Management Journal*, 49(1):69–73.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

B Hu, A Bajracharya, and H Yu. 2020. Generating medical assessments using a neural network model: Algorithm development and validation. *JMIR Medical Informatics*, 8(1):e14971.

Julia Ive, Natalia Viani, Joyce Kam, Lucia Yin, Somain Verma, Stephen Puntis, Rudolf N Cardinal, Angus Roberts, Robert Stewart, and Sumithra Velupillai. 2020. Generation and evaluation of artificial mental health records for natural language processing. *NPJ digital medicine*, 3(1):69.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Leo Anthony Celi, Roger Mark, and Steven Horng. 2023a. Mimic-iv-ed. PhysioNet.

Alistair Johnson, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. 2023b. Mimic-iv-note: Deidentified free-text clinical notes. PhysioNet.

Jean Kaddour, Joshua Harris, Maximilian Mozes, Herbie Bradley, Roberta Raileanu, and Robert McHardy. 2023. Challenges and applications of large language models. *arXiv preprint arXiv:2307.10169*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

SM Meystre, GK Savova, KC Kipper-Schuler, and JF Hurdle. 2008. Extracting information from textual documents in the electronic health record: a review of recent research. *Yearbook of Medical Informatics*, pages 128–144.

Amanda J Moy, Jessica M Schwartz, RuiJun Chen, Shirin Sadri, Eugene Lucas, Kenrick D Cato, and Sarah Collins Rossetti. 2021. Measurement of clinical documentation burden among physicians and nurses using electronic health records: a scoping review. *Journal of the American Medical Informatics Association*, 28(5):998–1008.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Xiangru Tang, Anni Zou, Zhuosheng Zhang, Yilun Zhao, Xingyao Zhang, Arman Cohan, and Mark Gerstein. 2023. Medagents: Large language models as collaborators for zero-shot medical reasoning. *arXiv preprint arXiv:2311.10537*.

Aykut Uslu and Jurgen Stausberg. 2021. Value of the Electronic Medical Record for Hospital Care: Update From the Literature. *Journal of medical Internet research*, 23(12).

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, Eduardo Pontes Reis, Anna Seehofnerova, et al. 2023. Clinical text summarization: Adapting large language models can outperform human experts. *Research Square*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Chaoyi Wu, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2023. Pmc-llama: Further finetuning llama on medical papers. *arXiv preprint arXiv:2304.14454*.

Justin Xu, Andrew Johnston, Zhihong Chen, Louis Blankemeier, Maya Varma, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and “discharge me!”. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. 2023. Fingpt: Open-source financial large language models. *arXiv preprint arXiv:2306.06031*.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10(1):586.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *arXiv preprint arXiv:2305.16739*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

A Extraction Method Paradigm

The extraction of the sections proves challenging. It first requires an iterative identification of the different section headers as shown in the algorithm 2. We perform this run on a subset of the discharges only and hence we may have missed some header. As an example here are the different headers we find for the section *Discharge Medications*: `['Discharge Medications:', 'Discharge medications:', '___ Medications:', '___ medications:']` ; here the *Section Basic Name* is `['Discharge Medications:'` and is the first header we consider for this section.

Algorithm 2 Algorithm for section header identification

```
Input: A section
section_headers ← [Section Basic Name]
found ← False
for discharge in all_discharges do
    for header in section_headers do
        if header in discharge then
            found ← True
            break
    if not found then
        Manually look for a new section header
        if new_header found then
            Add new_header to section_headers
    found ← False
return section_headers
```

B Direct Copy Examples in Summary

The examples provided below demonstrate how some text from the original raw sections is integrated in the BHC with minimal to none modifications. The raw text included for both examples are taken from the History of present illness section.

Example 1:

Raw text: This is a ___ yo f with h/o recently diagnosed metastatic cancer of unknown prior presenting with nausea, vomiting, and fever to 101 today.

Ground Truth BHC: ___ yo f with h/o recently diagnosed metastatic cancer of unknown primary presenting with nausea, vomiting, and fever to 101 on day of admission.

Example 2:

Raw text: ___ with HTN, HLD, & recurrent SVT on Flecainade/Toprol p/w CP/SOB and lightheadedness, found to be hypotensive with intermittent SVT without ischemic EKG changes or positive biomarkers, now admitted to the CCU for planned EP ablation.

Ground Truth BHC: ___ with HTN, HLD, & recurrent SVT on Flecainade/Toprol who presented with CP/SOB and lightheadedness, found to be hypotensive with intermittent SVT without ischemic EKG changes or positive biomarkers, admitted to the CCU for planned EP ablation.

C Full Tables of Direct Reference

D Dynamic Section Selection Algorithm

Extracting all sections enables intentional selection of sections for inclusion in our input, promoting consistency. By choosing a predefined set of sections, we ensure adherence to a standardized order (as shown in Fig.1) and consistent headers, which contrasts with the inherent variability of raw text. Furthermore, it establishes a consistent method for indicating missing sections: using

Algorithm 3 Algorithm for dynamic section selection

```
Input: All extracted sections, Section importance list, max length
$extract \leftarrow$ All extracted sections
$importance \leftarrow$ Section importance list
$max \leftarrow$ max length
Tokenize each section in $extract$
$total\_Length \leftarrow$ sum of tokenized section lengths in $extract$
if $total\_Length \leq max$ then
    return $extract$
else
    return TRY($extract$, $importance$, $max$, [])
procedure TRY($Allsections$, $importanceList$, $max\_length$, $removedSoFar$)
    for $i$ = length of $importanceList - 1$ to 0 step $-1$ do
        $currentSection \leftarrow importanceList[i]$
        $newRemovedList \leftarrow removedSoFar + [currentSection]$
        $remainingSections \leftarrow Allsections$ excluding $newRemovedList$
        $newTotalLength \leftarrow$ sum of tokenized section lengths in $remainingSections$
        if $newTotalLength \leq max\_length$ then
            return $remainingSections$
        TRY($Allsections$, $importanceList[0:i]$, $max\_length$, $newRemovedList$)
```

Section	**ROUGE-2**
History of Present Illness	0.01967
Discharge Condition	0.01263
Discharge Diagnosis	0.00939
Discharge Medications	0.00777
Pertinent Results	0.00673
Chief Complaint	0.00613
Past Medical History	0.00274
Physical Exam	0.00270
Major Surgical Procedures	0.00217
Medication on Admission	0.00173
Family History	0.00089
Discharge Disposition	0.00029
Allergies	0.00005
Social History	0.00004
Facility	0.00000
Service	0.00000
Sex	0.00000

Table 7: DI Direct Reference

'Header:\nNone\n' instead of various representations like '___', empty spaces, or simply the absence of the header commonly found in raw inputs. We then create our input by concatenating the desired sections. Even if a section is not chosen for inclusion in specific samples but was generally included for the subsequent experiment (like we do in strategy selection), 'Section Header:\nNone\n' is still included at the right spot to maintain consistency in input structure.

E Human Evaluation Criteria

The details of human evaluation criteria are here.

Section	**ROUGE-2**
History of Present Illness	0.08329
Pertinent Results	0.02621
Past Medical History	0.01515
Physical Exam	0.00702
Major Surgical Procedures	0.00575
Chief Complaint	0.00538
Family History	0.00180
Social History	0.00011
Allergies	0.00004
Sex	0.00000
Service	0.00000

Table 8: BHC Direct Reference

- **Completeness (captures important information)**
    - Captures no important information (1)
    - Captures ~25% of the important information (2)
    - Captures ~50% of the important information (3)
    - Captures ~75% of the important information (4)
    - Captures all of the important information (5)
- **Correctness (contains less false information)**
    - Contains harmful content that will definitely impact future care (1)
    - Contains incorrect content that is likely to impact future care (2)

  - Contains incorrect content that may or may not impact future care (3)
  - Contains incorrect content that will not impact future care (4)
  - Contains no incorrect content (5)

- **Readability**
  - Significantly harder to read than the reference text (1)
  - Slightly harder to read than the reference text (2)
  - Neither easier nor harder to read than the reference text (3)
  - Slightly easier to read than the reference text (4)
  - Significantly easier to read than the reference text (5)

F Generated Examples

BHC Generated Example

Ground Truth: The patient presented to the emergency department and was evaluated by the orthopedic surgery team. The patient was found to have a right femoral neck fracture. Given his progressive pain, decreased functionality, and the operative nature of his injury, L discussion was had with the patient using an interpreter regarding the risk and benefit of the surgery. After discussing the operation with his family he understood not only the risks, but also the benefits that he could expect from this operation. He was then admitted to the orthopedic surgery service. The patient was taken to the operating room on ___ for closed reduction and percutaneous pinning of the right femoral neck, which the patient tolerated well. For full details of the procedure please see the separately dictated operative report. The patient was taken from the OR to the PACU in stable condition and after satisfactory recovery from anesthesia was transferred to the floor. The patient was initially given IV fluids and IV pain medications, and progressed to a regular diet and oral medications by POD#1. Despite narcotic medications being made available to him, he manage his pain control with only acetaminophen. The patient was given ___ antibiotics and anticoagulation per routine. The patient's home medications were continued throughout this hospitalization. The patient worked with ___ who determined that discharge to **** was appropriate. The ___ hospital course was otherwise unremarkable.

Generated Text: The patient presented to the emergency department and was evaluated by the orthopedic surgery team. The patient was found to have a right femoral neck fracture and was admitted to the orthopedic surgery service. The patient was taken to the operating room on ___ for closed reduction and percutaneous pinning of his right femoral neck, which the patient tolerated well. For full details of the procedure please see the separately dictated operative report. The patient was taken from the OR to the PACU in stable condition and after satisfactory recovery from anesthesia was transferred to the floor. The patient was initially given IV fluids and IV pain medications, and progressed to a regular diet and oral medications by POD#1. The patient's home medications were continued throughout this hospitalization with exception of his home dose of prednisone, which he will resume upon discharge as he is no longer in-house. The ___ hospital course was otherwise unremarkable with no acute events occurring during this admission outside of his operation.

DI Generated Example

Ground Truth: Dear Ms. ___, It was a pleasure taking care of you at ___. You came to us because of shortness of breath which you had been intermittently experiencing since ___. Your blood tests did not show anything concerning for active ischemia. Your exam was not suggestive of acute heart failure. We suspect that your shortness of breath is related to a leaky valve and that improving blood flow through the arteries of the heart with a medication called isosrbide mononitrate will improve muscular function and the leakiness of the valve. You should take this medication daily. Please be sure to keep your upcoming appointment with Dr. ___ week. We noticed that your blood pressure has been high while you were here with us. We restarted your home lisinopril, now that your kidney function has improved back to normal. Sincerely, Your ___ Care Team.

Generated Text: Dear Ms. ___, You were admitted to the hospital because you were having shortness of breath. We did a stress test which was normal and did not show any signs of worsening heart disease. We also did an ultrasound of your heart which showed that your heart is pumping normally. We think that your shortness of breath may be related to narrowing in the arteries in your legs or to spasm in the arteries in your legs or lungs. For this we started you on a new medication called Imdur (isosorbide mononitrate) which can help with these symptoms. Please follow up with Dr ___ as listed below.

Models	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	METEOR	AlignScore	MEDCON	Overall
Llama2-7b-2k	0.025	0.287	0.065	0.131	0.218	0.209	0.211	0.582	0.216
Meditron-7b-2k	0.050	0.342	0.118	0.196	0.323	0.207	0.253	0.639	0.266
Meditron-7b-2k-s	0.069	0.385	0.146	0.227	0.367	0.241	0.287	0.670	0.299
Meditron-7b-d6k	0.059	0.376	0.141	0.225	0.370	0.214	0.280	0.663	0.291
Meditron-7b-l6k	0.080	0.398	0.153	0.230	0.376	0.264	0.304	0.683	0.311
Meditron-7b-l6k-s	**0.082**	**0.405**	**0.157**	**0.235**	**0.384**	**0.266**	**0.308**	**0.690**	**0.316**

Table 9: Global Models Results on the full Test Phase II set

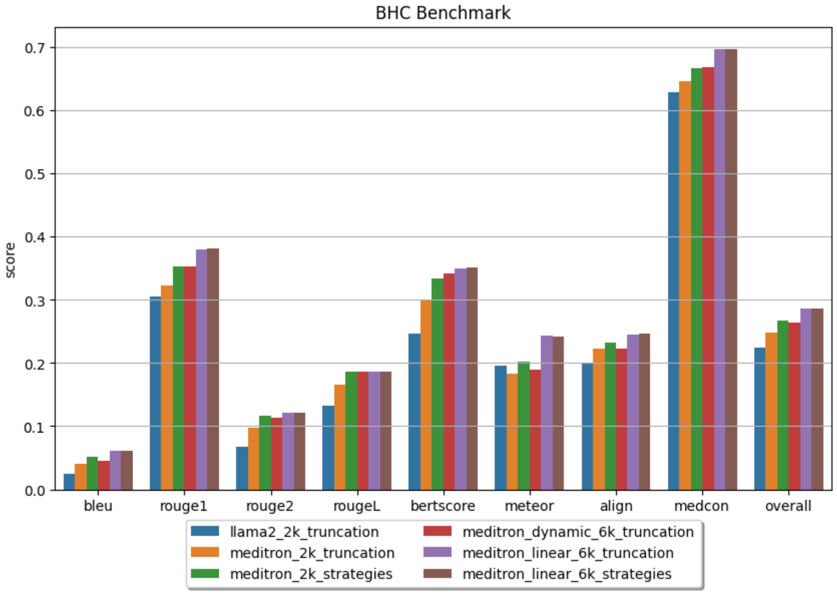


Figure 8: BHC Results on the full Test Phase II set

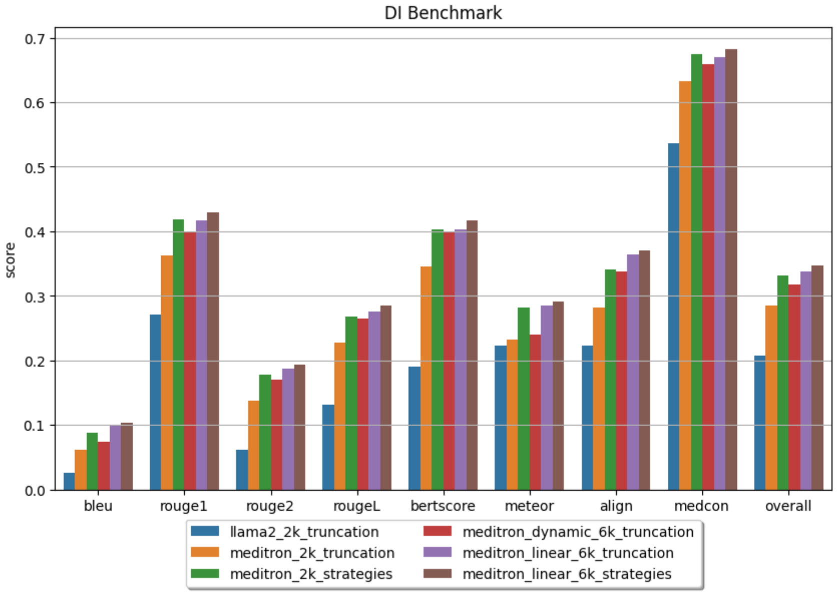


Figure 9: DI Results on the full Test Phase II set

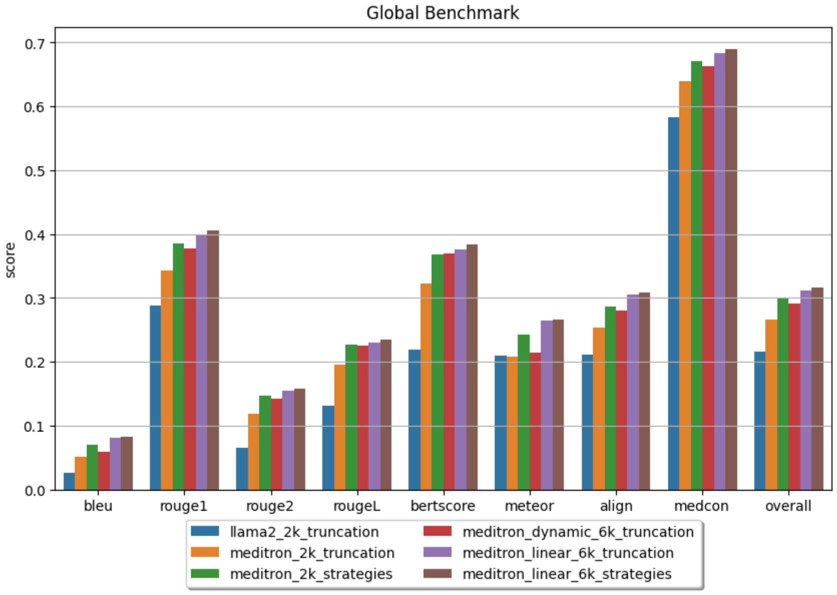


Figure 10: Global Results on the full Test Phase II set

UoG Siephers at "Discharge Me!": Exploring Ways to Generate Synthetic Patient Notes From Multi-Part Electronic Health Records

Erlend Frayling, Jake Lever and **Graham McDonald**
University of Glasgow
Scotland, UK

firstname.lastname@glasgow.ac.uk

Abstract

This paper presents the UoG Siephers team participation at the Discharge Me! Shared Task on Streamlining Discharge Documentation. For our participation, we investigate appropriately selecting and encoding specific sections of Electronic Health Records (EHR) as input data for sequence-to-sequence models, to generate the *discharge instructions* and *brief hospital course* sections of a patient's EHR. We found that, despite the large volume of disparate information that is often available in EHRs, selectively choosing an appropriate EHR section for training and prompting sequence-to-sequence models resulted in improved generative quality. In particular, we found that using only the *history of present illness* section of an EHR as input often led to better performance than using multiple EHR sections.

1 Introduction

In the clinical domain, writing notes about patients' health, diagnoses, and treatments is a necessary part of the patient healthcare journey, but it is also time consuming (Weiner and Biondich, 2006; Sinsky et al., 2016). The time spent by essential care staff, such as doctors and nurses, writing the notes in Electronic Health Records (EHRs) could be time better spent performing important clinical duties.

The *Discharge Me!* BioNLP ACL'24 Shared Task on Streamlining Discharge Documentation challenged participants to produce a system that can automatically generate: discharge instructions, which contain detailed guidelines provided to a patient upon their discharge from hospital; and Brief Hospital Courses, which summarise the key events, treatments and progress for a patient during their hospital stay (Xu et al., 2024). Discharge Me! participants were provided a dataset curated from the MIMIC-IV database (Johnson et al., 2023), which contains de-identified patients' EHRs.

EHRs are complex collections of, often long and disparate, reports about a patient's stay in hospital, including reports on patient demographics, medical history, laboratory tests and results, instructions for the patient and many more sections. In this work, we investigate several ways of appropriately selecting and encoding specific sections of EHRs as input data for sequence-to-sequence (seq2seq) models to generate the two target sections of the Discharge Me! shared task, i.e., the *discharge instruction* and the *brief hospital course*. In particular, in this work we investigate the following three research questions that guide our experimentation:

RQ1: What is the effect of using different sections of EHRs as training data for seq2seq models?

RQ2: Can a model that uses multiple EHR sections as input achieve better performance than models trained on single sections of EHRs?

RQ3: When concatenating multiple EHR sections as input, is it better to concatenate lexically, or concatenate embeddings post-encoding?

2 Related Work

Most relevant to our work is that of Hartman and Campion (2022), who employed various encoder-decoder models with different pre-trained checkpoints (Rothe et al., 2020) to generate a brief hospital course. Hartman and Campion attempted to summarise EHR records as short day-by-day summaries so that the EHR summaries would fit within the context limit of seq2seq encoder-decoder models. In our work, instead of summarising the input data to fit the context limit of an encoder-decoder model, we experiment with selectively choosing individual subsections of the EHR records to train seq2seq models.

Pal *et al.* (Pal et al., 2023) used the *nursing report* section of EHRs to generate a variety of EHR sections, such as the *history of present illness* and *discharge instructions*. The authors showed that

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 712–718
August 16, 2024. ©2024 Association for Computational Linguistics

seq2seq models, such as T5 and BART, can be effective for this task (Raffel et al., 2020; Lewis et al., 2019). Differently from Pal *et al.*, we explore using multiple sections of the EHR as input data, and ways to combine EHR sections as input.

Finally, the work of Liu et al. (2022) used the discharge instructions of historic patients, who had similar symptoms to a new patient, to write the new patient's discharge instruction. They used graph-based reasoning to generate the new discharge instruction based on the historic patients' instructions. Differently, we focus on using information that is entirely available in the patients own record and do not rely on the information of other patients.

3 Methods

In this section we describe our approaches for generating discharge instruction and brief hospital course sections of EHRs. Using different pre-trained encoder and/or decoder models within seq2seq models has been shown to be an effective way to adapt such models for different tasks (Rothe et al., 2020; Hartman and Campion, 2022). Therefore, in this work, we investigate three approaches for constructing the input data for seq2seq models, such that we can use the models' limited context effectively to model the EHRs sections. For each of our approaches, we deploy encoder-decoder models following the work of Hartman and Campion (Hartman and Campion, 2022).

3.1 Separate Text Sections

Our first approach uses individual EHR sections as the input to the seq2seq model. By using a specific self-contained section, we ensure that the training data is a focused and coherent report about the patient's medical history. In our experiments, we compare the effectiveness of two separate EHR sections, namely: History of Present Illness (HPI), which contains information about patients' stays in hospital; Radiology Report (RR), which contains information about patients' radiology exams.

We choose to evaluate the HPI and RR sections since the HPI sections encompass a lot of information that is also discussed in other sections of EHRs. Therefore, HPI sections can act as a general overview of the patient's condition, their reason for visiting the hospital, and their care plan. Differently, the RR section details specific observations about the physical condition of the patient. Indeed, there is little intersection between the information that is contained in the HPI and RR sections. Therefore, evaluating the sections separately can provide valuable insights about what kinds of information are most useful for automatically generating discharge instructions and brief hospital courses.

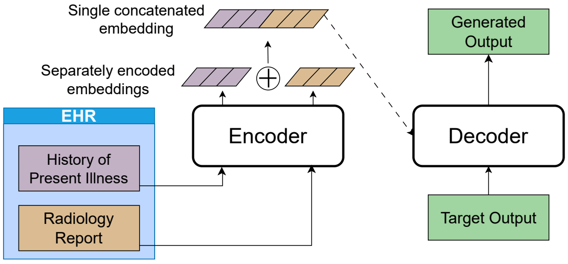

Single concatenated embedding
Separately encoded embeddings
EHR
History of Present Illness
Radiology Report
Encoder
Decoder
Generated Output
Target Output

Figure 1: Two EHR sections (purple and orange) are passed to the encoder separately, then their separate embedded representations, of corresponding colour, are concatenated before being passed to the decoder model.

3.2 Concatenating Text Sections

Secondly, we consider that multiple EHR sections may contain information that is essential for generating a correct discharge instruction or brief hospital course. In such a case, the approach presented in Section 3.1 would not be able to model all of the required information, due to the limited context of seq2seq models. Therefore, in this approach, we concatenate the text of the HPI and RR sections and use an encoder model that accepts longer context, i.e., Longformer (Beltagy et al., 2020).

3.3 Concatenating Embedded Sections

Finally, instead of concatenating the *text* of the different EHR sections, as described in Section 3.2, in this approach we encode the HPI and RR sections separately and then concatenate the *encoded* sections, before passing the concatenated encodings to the cross attention layers of the decoder model. This approach is inspired by how the reader component of Fusion-in-Decoders (Izacard and Grave, 2020) performs question answering tasks with multiple retrieved context documents. We therefore refer to this approach as our fusion approach.

Figure 1 illustrates our fusion approach. In this approach, a model with a shorter context limit can be used, though encoding the multiple sections *separately* increases computation time linearly. Moreover, the sections to be encoded are distinct, separate reports. Concatenating the sections into a single long passage to encode, such as in Section 3.2, may result in a low-quality embedded representation that cannot capture the diversity of the different textually concatenated reports. In this approach, by encoding EHR sections separately, the contextual separation is retained for each section and we hypothesise that this may improve performance in the overall seq2seq generation task.

4 Experimental Setup

In this section, we present the experimental setup to investigate the three research questions that we presented in Section 1.

4.1 Dataset

The dataset for the task was provided by the organisers of the Discharge Me! shared task and is curated from the MIMIC-IV database. The data can be downloaded from Physionet.[1] The dataset is split into a training set (68,785 samples), a validation set (14719 samples), a phase 1 testing set (14702 samples) and a phase 2 testing set (10962 samples). Each sample corresponds to an emergency department admission with an associated discharge summary, which contains many different reports on a patient's stay in hospital. Each sample also contains at least one RR. Finally, each sample also contains a discharge instruction and a brief hospital course section, which are the two target sections to be generated. The dataset includes gold standard discharge instruction and brief hospital course sections for each of the training, validation, phase 1 testing and phase 2 testing sets.

For our approaches described in Section 3, we extract the HPI section and the most recent RR section (some samples contain more than one RR section) for each sample in the dataset using Python Regular Expressions. From early exploratory work, we discovered that, for the models that we evaluate, using a large number of samples for training offers little performance improvements compared to using a smaller subset of the data. We therefore use a subset dataset of 5000 random samples from the training set, and 1000 random samples from the validation set to train our chosen models.

4.2 Models

We now provide a description of the different models and model architectures that we deploy in our experiments. In all cases we train two versions of each model. One version is trained to generate the target discharge instruction, while the other is trained to generate the target brief hospital course. In all cases, we train the models using the HPI and/or RR sections to generate the target sections.

Firstly, to investigate the approach presented in Section 3.1, we evaluate several encoder-decoder seq2seq models that are trained on a single input, either HPI or RR, and leverage pre-trained checkpoints following Rothe *et al.* (Rothe et al., 2020). We deploy three encoder models: a RoBERTa encoder (Liu et al., 2019), since it was found to be the best performing by Rothe *et al.* (Rothe et al., 2020); the ClinicalBERT encoder (Alsentzer et al., 2019), as it is pre-trained on MIMIC data; and the BERT encoder (Devlin et al., 2018) as an appropriate baseline. We deploy the same decoder, GPT-2 (Radford et al., 2019), in all instances. Additionally, we also deploy a base-size T5 model (Raffel et al., 2020) since it has been shown to be effective for seq2seq tasks. Our participation in the Discharge Me! shared task investigated the effectiveness of different encoder-decoder models, however for completeness we deploy two decoder-only models, namely GPT-2 (Radford et al., 2019) and Llama 3 8B (Meta, 2024). We train the decoder-only models by passing the input and target sections as one string, where the sections are separated by three newline characters. At inference time the models are passed only the input data and newline characters.

For our approach that we described in Section 3.2, we deploy an encoder-decoder model with a pre-trained Longformer encoder model (Beltagy et al., 2020) and GPT-2 decoder. The Longformer model has a context window of up to 4096 tokens, ensuring that for EHR samples in our dataset the HPI and RR sections can be concatenated before encoding, as described in Section 3.2. We concatenate the sections as separate paragraphs by joining the sections with connecting newline characters.

Finally, for our fusion approach, presented in Section 3.3, we deploy two encoder-decoder models. Both models use a RoBERTa encoder and GPT-2 decoders. One model uses the base size GPT-2 decoder, the other model uses GPT-2 large. We refer to these models as Fusion-roBERTa-GPT2 and Fusion-roBERTa-GPT2-large respectively.[2]

For all of our models, we perform 15 runs of hyperparameter tuning using Optuna (Akiba

[1] https://physionet.org/

[2] We also submitted BERT-GPT2 and ClinicalBERT-GPT2 fusion runs to the Discharge Me! leader board. However, the official evaluation script was not able to generate results for us to evaluate these models.

Model	**Overall Score**	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	Meteor	AlignScore	MEDCON
				Encoder-Decoder Architectures					
T5 (HPI)	**0.191**	0.017	**0.341**	**0.108**	**0.209**	**0.268**	0.247	0.143	**0.193**
T5 (RR)	0.079	0.001	0.128	0.008	0.080	0.130	0.073	**0.157**	0.054
BERT-GPT2 (HPI)	0.144	0.012	0.258	0.045	0.124	0.245	0.242	0.113	0.114
BERT-GPT2 (RR)	0.156	0.011	0.294	0.055	0.157	0.253	0.244	0.105	0.128
roBERTa-GPT2 (HPI)	0.143	0.009	0.250	0.025	0.135	0.251	0.239	0.107	0.124
roBERTa-GPT2 (RR)	0.110	0.005	0.183	0.010	0.092	0.198	0.188	0.119	0.083
ClinicaBERT-GPT2 (HPI)	0.143	0.011	0.254	0.020	0.131	0.252	0.239	0.108	0.124
ClinicalBERT-GPT2 (RR)	0.149	0.012	0.272	0.046	0.144	0.252	0.240	0.106	0123
LongFormer-GPT2	0.152	0.013	0.278	0.030	0.153	0.255	0.244	0.110	0.135
Fusion-roBERTa-GPT2	0.148	0.011	0.264	0.029	0.137	0.255	0.243	0.113	0.130
Fusion-roBERTa-GPT2-large	0.159	**0.039**	0.222	0.042	0.146	0.251	**0.266**	0.134	0.169
				Decoder-only Models					
GPT-2 (HPI)	0.153	0.009	0.284	0.035	0.139	0.241	0.206	0.167	0.151
GPT-2 (RR)	0.128	0.011	0.160	0.021	0.101	0.232	0.212	0.158	0.131
Llama 3 (HPI)	<u>0.196</u>	<u>0.028</u>	<u>0.350</u>	<u>0.091</u>	<u>0.180</u>	<u>0.300</u>	<u>0.218</u>	<u>0.172</u>	<u>0.230</u>
Llama 3 (RR)	0.168	0.016	0.322	0.072	0.160	0.264	0.195	0.151	0.167

Table 1: Results for our different methods evaluated by the official Discharge Me! submission system. **Bold text** indicates the best scoring encoder-decoder results. <u>Underlined text</u> indicates the best scoring decoder-only results.

et al., 2019), searching learning rate (1e-6 to 1e-3), weight decay (1e-4 to 1e-2), and number of epochs (1 to 9). We optimise for evaluation loss and use the best hyperparameter configuration to train a final model that is used in evaluation, all using 3 NVIDIA RTX A6000 GPUs. To fine-tune the Llama 3 model we use gradient accumulation (Goodfellow et al., 2016; Bengio, 2012), processing batches of 4 and accumulating to batches of 8, 8 being the batch size we use to train all of our other models. We also use Quantised Low Rank Adaptation to fine-tune the Llama 3 model (Dettmers et al., 2024)

4.3 Evaluation Metrics

The two generated target texts of each model are evaluated independently against their corresponding gold standard texts using a variety of text-based similarity metrics and factual correctness metrics. The metrics used for evaluation are: BLEU-4 (Papineni et al., 2002); the ROUGE metrics including ROUGE-1, ROUGE-2 and ROUGE-L (Lin, 2004); BertScore; Meteor (Banerjee and Lavie, 2005); AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023). Each of the eight metric scores for each of the two generated datasets are then averaged to get combined score for each metric, and then finally all eight scores are average again to produce a single *overall score*.

5 Results

This section describes our findings relating to the research questions presented in Section 1. Table 1 provides the results our models achieved when submitted to Discharge Me! leaderboard (Xu et al., 2024). Overall for the encoder-decoder models, a T5 model trained on HPI sections of patient EHRs was the best performing model, achieving 0.191 Overall Score, with the next best approach BERT-GPT2(RR) achieving 0.156 Overall Score. Additionally, the Llama 3 decoder-only model achieves competitive performance with the T5 model when using the HPI sections of EHRs. This is, arguably, to be expected given the much larger size, and recency of the model. Furthermore, both decoder-only models, Llama 3 and GPT-2, perform better when using the HPI as input rather than the RR section. This is in line with our findings for encoder-decoder models.

Concerning RQ1, the T5 and RoBERTa-GPT2 models perform better when trained on the HPI input. On the other hand, the BERT-GPT2 model and ClinicalBERT-GPT2 model perform better when trained with RR input. However, the performance increases that are obtained from training on HPI data are notably greater than any performance improvements that are obtained from training on RR data. The T5(HPI) model shows 141% improvement in Overall Score compared to the T5(RR) model, whereas the BERT-GPT2(HPI) model resulted in only a 7% Overall Score drop compared to its BERT-GPT2(RR) counterpart. Similarly, roBERTa-GPT2(HPI) achieves a 29% improvement over roBERTa-GPT2(RR), while there is only a 4% drop in Overall Score between ClinicalBERT-GPT2(RR) and ClinicalBERT-GPT2(HPI). Answering RQ1, we find that when training on individual record sections, training on HPI most often leads to better performance compared to models trained on RR. Indeed, the gains in Overall Score from training on HPI compared to RR are notably greater.

In answering RQ2, we find that training seq2seq models on multiple concatenated sections of EHR with models does not outperform models trained on a single input section of a record. Our best performing concatenation model, LongFormer-GPT-2, outperforms several models trained on single EHR sections. However, both BERT-GPT2(RR) and T5(HPI) both outperform the Longformer-GPT2 model. This indicates that choosing a single input section for the *right* model can outperform a model that has access to both sections of the data. Specifically, we see that the Longformer-GPT2 outperforms the BERT-GPT2(RR) model by a small margin in BERTScore and Meteor. However, the two models perform very similarly, indicating that the additional information in HPI that was available to the Longformer-GPT2 model did not improve its performance markedly. Thus, training on additional information is not always beneficial.

Regarding RQ3, we find that concatenating the different sections of EHR records lexically, and then using an encoder with a larger context window is a more effective method for this task than encoding the different sections separately as proposed in Section 3.3. Neither of our fusion models beat the LongFormer-GPT2 model in overall score, despite Fusion-roBERTa-GPT2-large using a larger decoder model. Considering this in the context of the findings of our first and second research question, this may indicate that the decoder model is not able to utilise the separately embedded records sections as effectively as it is able to understand embeddings of a single section of the report. Replacing the decoder with a larger model does improve performance, but still the performance is worse than a T5-base model trained on the single HPI section.

6 Qualitative Analysis

In this section we investigate the generated record quality for the best performing seq2seq model, T5(HPI). We analyse the ten highest and ten lowest scoring generated discharge instructions and brief hospital course, in terms of their ROUGE-1 scores.

In the highest scoring generated EHR sections, the core ailments of the patients are correctly described. In the generated discharge instructions, the recommended followup treatment is often inaccurate but the structure of the instructions, which all contain subheadings (e.g. "why you were in hospital", "what you should do after leaving"), are usually correctly generated and match the target texts. This improves the overall quality of the generated discharge instructions. The generated brief hospital courses match most of the text of their corresponding target texts exactly. However, they deviate towards the end of the text often adding extended information that is still topically relevant, but not actually part of the true target text.

Inspecting the lowest scoring generated samples, we find common problems with the generation process for both the discharge instructions and brief hospital courses. While our models are effective at writing structured discharge instructions with specific sub headings, and brief hospital courses that contain a verbose description of the patient's problems, the effectiveness of the generation degrades when the target texts are not in line with these formats. For example, when the target discharge instruction is a short single-line note, such as instructions about avoiding a certain kind of food, or a reminder for the patient to weigh themselves, the models attempts to generate a long instruction with many unnecessary subsections. Similarly, our model will attempt to generate a verbose brief hospital course, even when the true target is a list of vital-sign readings. Uniquely to the discharge instruction generation, we find that several of the target sections are written in Spanish. In such cases, our model still attempts to generate English text, as the input section is always written in English.

7 Conclusion

To conclude, we have found that training a seq2seq model to generate discharge instructions and brief hospital courses using single sections from Electronic Health Records (EHR) as input, outperformed models trained using multiple sections of EHR as input. Moreover, choosing which single section to use as input is an important factor that depends on the chosen seq2seq model and that generally, some sections can expect to provide reasonable performance overall compared to others.

Acknowledgments

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) grant number EP/X018237/1.

8 Limitations

We note that there are many potential extension to our experiments that could provide additional valuable insights beyond the scope of this work. Firstly, in our work we use only a GPT-2 decoder in all our encoder-decoder models, while in Table 1 we find that a Llama 3 decoder-only model outperforms the GPT-2 decoder-only model. Therefore, we could, for example, evaluate Llama 3 as the decoder in an encoder-decoder architecture. Moreover, evaluating different sizes of decoder models would also bring additional insights. For example, the results for the Fusion-roBERTa-GPT2-large model in Table 1 show that using a larger variant of GPT-2 decoder in the encoder-decoder architecture improves overall performance.

Secondly, in our work we only investigate using two sections of EHRs, namely the History of Present Illness section and the Radiology Report. Though ultimately we found using one of these two sections to train a model was more effective than combining both sections as input, further research to explore the use of other sections of the EHR poses interesting questions. For example, are there other sections that are better to use as input than the ones we have chosen to use? Moreover, concerning the approaches described in Sections 3.2 and 3.3, how does increasing the number of EHR sections that are concatenated as input affect performance?

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631.

Emily Alsentzer, John R Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical bert embeddings. *arXiv preprint arXiv:1904.03323*.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *CoRR*, abs/2004.05150.

Yoshua Bengio. 2012. Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade*, pages 437–478. Springer.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2024. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep learning*. MIT press.

Vince Hartman and Thomas R Campion. 2022. A day-to-day approach for automating the hospital course section of the discharge summary. *AMIA Summits on Translational Science Proceedings*, 2022:216.

Gautier Izacard and Edouard Grave. 2020. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*.

Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. 2023. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Fenglin Liu, Bang Yang, Chenyu You, Xian Wu, Shen Ge, Zhangdaihong Liu, Xu Sun, Yang Yang, and David Clifton. 2022. Retrieve, reason, and refine: Generating accurate and faithful patient instructions. *Advances in Neural Information Processing Systems*, 35:18864–18877.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Meta. 2024. Introductng meta llama 3: The most capable openly available llm to date.

Koyena Pal, Seyed Ali Bahrainian, Laura Mercurio, and Carsten Eickhoff. 2023. Neural summarization of electronic health records. *CoRR*, abs/2305.15222.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. 2020. Leveraging pre-trained checkpoints for sequence generation tasks. *Transactions of the Association for Computational Linguistics*, 8:264–280.

Christine Sinsky, Lacey Colligan, Ling Li, Mirela Prgomet, Sam Reynolds, Lindsey Goeders, Johanna Westbrook, Michael Tutty, and George Blike. 2016. Allocation of physician time in ambulatory practice: a time and motion study in 4 specialties. *Annals of internal medicine*, 165(11):753–760.

Michael Weiner and Paul Biondich. 2006. The influence of information technology on patient-physician relationships. *Journal of general internal medicine*, 21(Suppl 1):35–39.

Justin Xu, Andrew Johnston, Zhihong Chen, Louis Blankemeier, Maya Varma, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10(1):586.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Roux-lette at “Discharge Me!”: Reducing EHR Chart Burden with a Simple, Scalable, Clinician-Driven AI Approach

S. Wendelken[*], A. Antony[**], R. Korutla[*], B. Pachipala[*], D. Mahajan[**], J. G. Shanahan[**], W. Saba[**]
[*]The Roux Institute at Northeastern University
[**]The Institute for Experiential AI at Northeastern University

Abstract

Healthcare providers spend a significant amount of time reading and synthesizing electronic health records (EHRs), negatively impacting patient outcomes and causing provider burnout. Traditional supervised machine learning approaches using large language models (LLMs) to summarize clinical text have struggled due to hallucinations and lack of relevant training data. Here, we present a novel, simplified solution for the “Discharge Me!” shared task. Our solution uses a question-based approach to treat this summarization task as a context-aware and domain-specific question-answering process. Our pipeline prompts an LLM answer *specific questions* posed by subject-matter experts (SMEs) using only patient specific context data. This method (i) avoids hallucinations through hybrid RAG/zero-shot contextualized prompting; (ii) requires no extensive training or fine-tuning; and (iii) is adaptable to various clinical tasks.

1 Introduction

Clinicians spend considerable amount of time navigating vast amounts of electronic health records (EHRs) information, often spending 2-3 times more time interacting with EHRs than with patients [1][4]. Much of this time is spent on manual information retrieval and stylized summarization of relevant content, referred to as the charting burden. The "Discharge Me!" shared NLP task [8][9] aims to alleviate this burden by automating the generation of "Brief Hospital Course" (BHC) and discharge instructions from EHRs in the MIMIC IV dataset [8]. These sections are critical but time-consuming, requiring synthesis, interpretation, and summarization of data from various parts of the patient’s medical charts include admission notes, progress notes, lab results, and radiology reports.

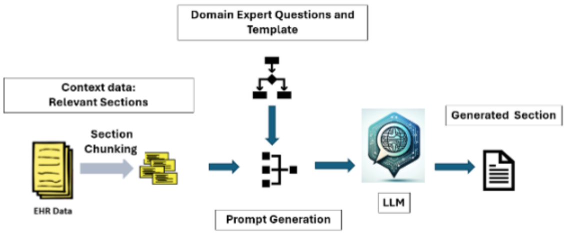

Domain Expert Questions and Template
Context data: Relevant Sections
Section Chunking
EHR Data
Prompt Generation
LLM
Generated Section

***Figure 1**: Simplified solution to “Discharge Me!” task.*

Automated approaches such as rule-based systems and supervised machine learning models have been explored but often struggle with the complexity and variability of clinical data. These challenges lead to issues like hallucinations—plausible but incorrect information—and incomplete summaries due to missing data in the training sets. Despite significant progress in machine learning for NLP tasks, achieving contextualized, relevant, and accurate summarization remains challenging. Supervised ML approaches face difficulties in domain adaptation, and fine-tuning does not fully address these issues [1][3][5]. Moreover, and while some progress was reported using fine tuning (e.g., [11]), the approach we describe here achieves better results without bearing the enormous cost of fine tuning. The lack of training data and LLMs' limitations in handling domain-specific data further complicate the problem. Additionally, different users require different types of information extracted from EHRs, making contextual summarization essential.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 719–723
August 16, 2024. ©2024 Association for Computational Linguistics

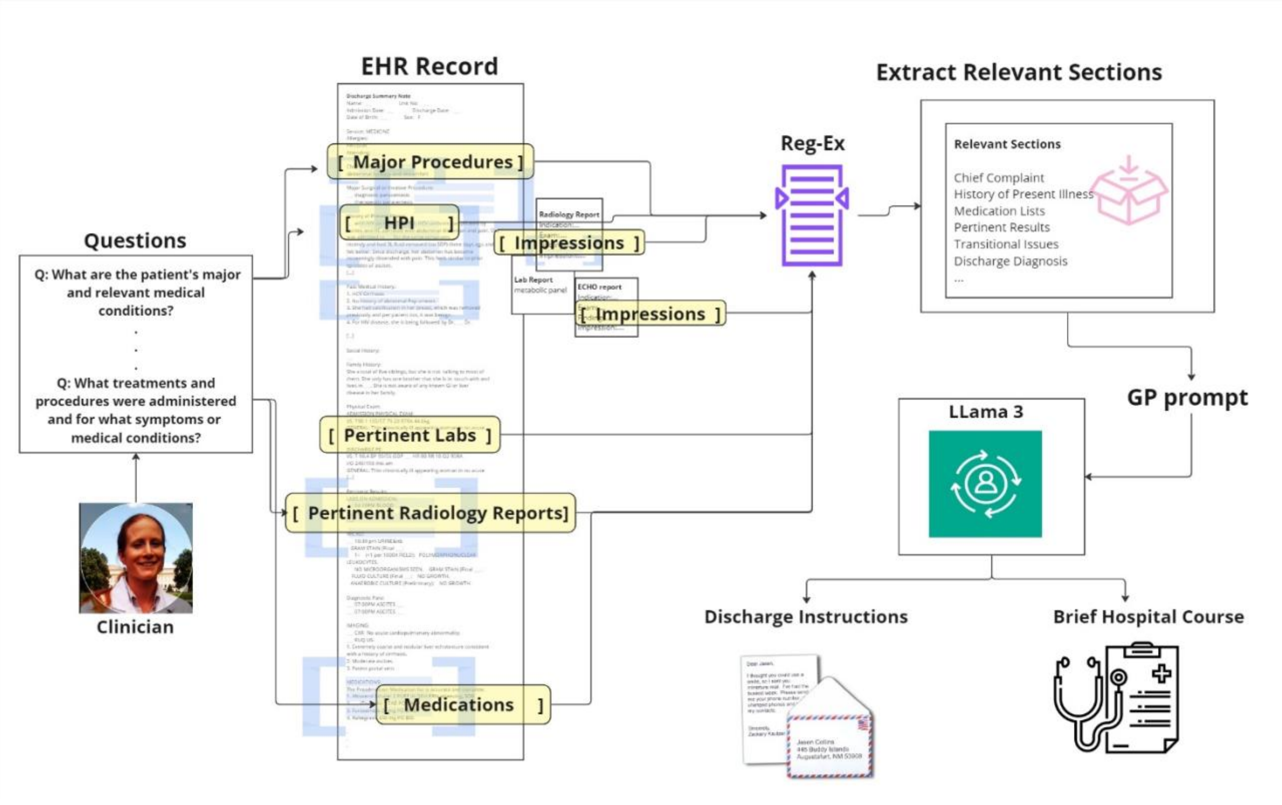


Figure 2: *"GP prompting" pipeline splits the EHR record into substring contexts to which domain expert questions are applied in the form of prompting. The responses to these questions are then used to generate a task centric response.*

Our solution uses a question-based approach to treat this summarization task as a context-aware and domain-specific question-answering process (see Figure 1). The summaries generated using our pipeline answer *specific questions* posed by subject-matter experts (SMEs). Previous experiments of our approach for lengthy clinical note summarization using general-purpose models like ChatGPT showed their ability to understand clinical terms and abbreviations without fine-tuning [6]. This supports our hypothesis that LLMs can efficiently generate accurate summaries without hallucinations when provided with the relevant contextual data. Our method is customizable, contextual, and bypasses the need for extensive training datasets. It harmonizes unstructured clinical data and can be applied to any domain requiring contextualized summaries for various users.

2 Methods

We developed an AI pipeline to automate clinical workflow tasks for generating discharge summaries, leveraging large language models (LLMs) to answer questions, retrieve information, infer actions, and summarize the information in a stylized format, where a domain expert guided the questions and output style (See Figure 2). We call our approach "GP-Prompting". Our first iteration used the general-purpose LLM Meta-Llama-3-8B-Instruct [10] installed locally on a A10G GPU.

The following is a summary of our pipeline:

1. *Data Sources*: A subset of the MIMIC IV data set including Discharge Summary text notes, radiology reports, and initial ICD diagnosis codes provided by the task organizers [8].
2. *Pre-processing*: Segmentation of discharge note into logical sections. This was done by extracting content by section, where section headings were identified from the dataset by the clinician. Section headers were not standardized in the data set and were thus matched to known variations using regexp. E.g., "History of Present Illness" | "HPI". Extraneous text (e.g., repeated "=") were removed
3. *"GP-prompting"*: Our method is a hybrid of question answering / zero-shot prompting.
   a. Domain Expert Input: questions formulated by clinician to identify key medical information.
   b. Select relevant sections of the clinical notes data that will contain answers to the

questions suggested by a domain expert. E.g. Select the "HPI" section to answer questions about the initial treatment course. Use the "Admission and Discharge Medications" lists to deduce what new medications the patient was prescribed for discharge. The selected text will be used as the context data.

c. Prompt Template: Prompts were constructed by concatenating a {Topic}, {General Instructions}, {Output Template}, {SME Questions}, {Text Generation Instructions}, {Context} data.

Topic: "The topic is about clinical notes, medical records, and other text documents from electronic health records (EHR) from a patient's hospital admission."

General Instructions: "You will be provided with input text and data from a specific patient's medical records. Use only this data to answer questions about the EHR.

"Discharge Instruction" generation

Output Template:

": "Dear ____, It was a pleasure to take care of you during your recent hospital admission. You were admitted to the hospital because [explain the reason for admission]. [Briefly explain the diagnostic work up and explain the results] During your hospital stay your treatments included [briefly explain the major treatments and procedures]. … Sincerely, Your Team"

SME Questions:

"Why was the patient admitted to the hospital?
What treatments and procedures were administered and for what symptoms or medical conditions?
What was the diagnostic work up related to the chief complaint? What were the notable results?
What medications were used to treat the patient? Were any medications new or discontinued?
Are there changes to the existing medications?
What are the ongoing issues and follow-up recommendations?"

Text Generation Instructions: ": Write a letter to the patient that summarizes their hospital stay and communicates follow-up instructions as well as important changes to their medications. Use the provided template and answers to the questions above to fill in the blanks.

Selected Context: 'CC', 'HPI', 'Transitional Issues', 'Discharge Diagnosis' from the discharge summary note.

Example "Discharge Instructions" output:

"Discharge Instructions ---hadm_id: 27729294

Dear ___, It was a pleasure to take care of you during your recent hospital admission. You were admitted to the hospital because of your leg swelling and complaints of oral ulcers. During your hospital stay, your treatments included antibiotics for cellulitis in your leg and a medication called Duonebs. You were diagnosed with cellulitis in your leg and were given a course of antibiotics to help clear up the infection. You were also treated for your bullous pemphigoid and lymphoma in your mouth. You will need to follow up with your doctor to monitor your condition and make sure that the infection has cleared up. You will also need to continue taking your medication as prescribed and follow up with your dermatologist to manage your bullous pemphigoid. If your symptoms return, please call our office (555-555-5555) or 911 for immediate assistance. Sincerely, Your Team---"

Brief Hospital Course (BHC) Generation:

Output Template: "___ with a history of [briefly describe the relevant medical history or existing medical problems] presents with [briefly summarize the patient's symptoms]. They were admitted to the [what unit, service or floor were they admitted to?] for [briefly summarize the reason for admission, initial diagnosis, and any major procedures or treatments]. [Briefly summarize the discharge diagnosis at discharge]

Problem-based hospital course—

# [Medical condition 1] --- [briefly describe presentation and severity for this medical condition or symptom]. [Summarize the related diagnostic work up and notable results]. [Summarize the related medical or surgical treatment course].

# [Medical condition 2] -- [briefly describe presentation and severity]..."

SME Questions for BHC:

"What are the patient's major medical conditions? Why was the patient admitted to the hospital? What symptoms did the patient present with? What was the severity and duration of the presenting symptoms? What treatments and procedures were administered and for what symptoms or medical conditions? What unit or floor were they admitted to? What was the diagnostic work up? What were the notable results? What medications were used to treat the patient? Were any medications new or discontinued? Are there changes to the existing medications? What was the patient diagnosed with at discharge? What was the patient initially diagnosed with?"

Text Generation Instructions: “Fill in the blanks: Use the answers to the questions above to fill in the blanks. If you do not know the answer, fill in the blank with "___". Answer clearly and concisely”
Context: text data included “Chief Complaint”, “HPI”, “Discharge Diagnosis,” and “Medication Lists” from the discharge note, and “ICD Diagnosis title” from the diagnosis table
Example BHC output for hadm_id 27729294: “The patient with a history of bullous pemphigoid and lymphoma presents with leg swelling and complaints of oral ulcers. They were admitted to the medical floor for evaluation of leg lesions and abdominal swelling, initially diagnosed with cellulitis of the left lower limb.

Problem-based hospital course---

# Cellulitis --- The patient presented with leg swelling and complaints of oral ulcers. The diagnostic work up included labs notable for WBC of 12.5, and imaging such as LLE US did not show evidence of DVT; CXR showed mild cardiomegaly and mild vascular congestion. The patient was given Duonebs and CTX 1g IV for LLE cellulitis. # Bullous pemphigoid --- The patient has a history of bullous pemphigoid and lymphoma in his mouth. The diagnostic work up included bedside US did not show any evidence of ascites.”

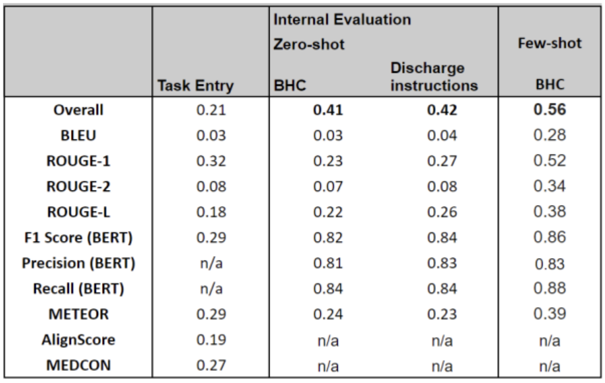
3 Evaluation

Outputs for all 10,962 targets were submitted to the shared task and evaluated against a hidden subset of 250 records by the task organizers as described in [9]. Results are presented in Table 1. Concurrently, we evaluated the first 250 generated outputs against human-generated targets using BLEU-4, ROUGE-1/-2/-L, BERTScore (Precision, Recall, F1), and METEOR scores. A clinician visually inspected the first 10 outputs for accuracy and recall during pipeline development.

Time and computational constraints limited our ability to fully optimize the pipeline during the contest. Post-contest, we tested a few-shot approach for a single record with five example prompt-data pairs, showing promising results for BHC generation.

4 Results

Initial results showed promising performance, though further optimization with few-shot learning and refined model parameters could improve accuracy and efficiency.



		Internal Evaluation Zero-shot		Few-shot
	Task Entry	BHC	Discharge instructions	BHC
Overall	0.21	**0.41**	**0.42**	**0.56**
BLEU	0.03	0.03	0.04	0.28
ROUGE-1	0.32	0.23	0.27	0.52
ROUGE-2	0.08	0.07	0.08	0.34
ROUGE-L	0.18	0.22	0.26	0.38
F1 Score (BERT)	0.29	0.82	0.84	0.86
Precision (BERT)	n/a	0.81	0.83	0.83
Recall (BERT)	n/a	0.84	0.84	0.88
METEOR	0.29	0.24	0.23	0.39
AlignScore	0.19	n/a	n/a	n/a
MEDCON	0.27	n/a	n/a	n/a

Table 1: *Scoring metrics for “Discharge Me!” generated outputs “Brief Hospital Course” and “Discharge Instructions”*

5 Discussion

Our approach demonstrated a simple and effective method for automatically generating the “Brief Hospital Course” and “Discharge Instructions” sections of discharge summary notes. Future improvements include integrating few-shot learning, fine-tuning, and principled chunking with retrieval-augmented generation (RAG). Experimentation with various LLM sizes and optimization of parameters (e.g., temperature, different values for *top_k*), topic tracking, and integration of structured chart data (not available for this task) can enhance output quality and speed.

6 Limitations

The task dataset was unrealistic, lacking essential components present of typical charts such as daily progress notes, procedure notes, labs, vitals, microbiology, radiology reports, and medication administration records (MAR). Generating the BHC and discharge instructions without comprehensive event data leads to hallucinations. The provided dataset, containing only discharge summary notes, is insufficient for accurate BHC or discharge instructions, especially for patients with extended hospital stays.

Additionally, the target dataset sections were often inaccurately segmented from the input. Approximately 16% of phase 2 BHC targets were severely incomplete, often under 100 words. In these cases, the extraction was truncated due to an unexpected heading, often missing the problem-based treatment course entirely. The targets also often incorrectly included content from the "Transitional Issues" section, which should be separate from the BHC.

We lacked comprehensive data such as daily progress notes and outpatient referrals, so we utilized selected parts of the discharge summary, including the HPI Medication list, which provided partial relevant information needed for the BHC. All selected input sections were considered by the clinician to be accessible during the typical clinical workflow. Incomplete records often resulted in outputs lacking the full content of the target data, but it was reassuring that the model did not hallucinate.

Pipeline Challenges: Due to data-use agreements, models and data had to be run locally and securely, necessitating downloaded LLMs. This limitation prevented the use of faster, publicly available pipelines, resulting in lower accuracy, and slower local model outputs compared to more advanced models that we plan to use in the future.

We also noted a discrepancy between contest and internal BERTScores. At the time of this publication, the root cause of this discrepancy is unknown, but it is likely resulting from using different BERTScore functions (we used a standard "bert_score" import, whereas the contest scoring used a custom BERTScore script). Similarly, the custom AlignScore and MEDCON scores used for contest were not implemented during our evaluation process as we were unable to successfully run the custom scripts in time for the contest entry.

7 Conclusion

The solution we presented was an efficient, context-aware, question-based approach to automate the generation of discharge summaries. Despite the constraints and limitations of the dataset and evaluation metrics, our method showed promise, particularly with a few-shot learning approach. Future work will focus on refining chunking methods for a RAG-based approach, optimizing prompts, and exploring various LLM configurations to improve accuracy and reliability in clinical settings.

References

[1] E. Hossain et al., "Natural Language Processing in Electronic Health Records in relation to healthcare decision-making: A systematic review," Comput. Biol Med., vol. 155, p. 106649, Mar. 2023, doi: 10.1016/j.compbiomed.2023.106649.

[2] T. Searle, Z. Ibrahim, J. Teo, and R. J. B. Dobson, "Discharge summary hospital course summarization of in-patient Electronic Health Record text with clinical concept guided deep pre-trained Transformer models," J. Biomed. Inform., vol. 141, p. 104358, May 2023, doi: 10.1016/j.jbi.2023.104358.

[3] D. Van Veen et al., "Clinical Text Summarization: Adapting Large Language Models Can Outperform Human Experts," Res. Sq., p. rs.3.rs-3483777, Oct. 2023, doi: 10.21203/rs.3.rs-3483777/v1.

[4] J. M. Overhage and D. McCallie, "Physician Time Spent Using the Electronic Health Record During Outpatient Encounters," Ann. Intern. Med., vol. 172, no. 3, pp. 169–174, Feb. 2020, doi: 10.7326/M18-3684.

[5] T. Searle, Z. Ibrahim, and R. J. Dobson, "Experimental Evaluation and Development of a Silver-Standard for the MIMIC-III Clinical Coding Dataset," in Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing, 2020, pp. 76–85. doi: 10.18653/v1/2020.bionlp-1.8.

[6] W. Saba, S. Wendelken, and J. Shanahan, "Question-Answering Based Summarization of Electronic Health Records using Retrieval Augmented Generation." Preprint https://arxiv.org/ftp/arxiv/papers/2401/2401.01469.pdf

[7] A. S. Afshar et al., "An exploratory data quality analysis of time series physiologic signals using a large-scale intensive care unit database," JAMIA Open, vol. 4, no. 3, p. ooab057, Jul. 2021, doi: 10.1093/jamiaopen/ooab057.

[8] J. Xu, "Discharge Me: BioNLP ACL'24 Shared Task on Streamlining Discharge Documentation." [object Object]. doi: 10.13026/4A0K-4360. https://physionet.org/content/discharge-me/1.2/

[9] J. Xu et al., "Overview of the First Shared Task on Clinical Text Generation: RRG24 and 'Discharge Me!,'" in The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks, Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024.

[10] AI@Meta, "meta-llama/Meta-Llama-3-8B-Instruct · Hugging Face." Accessed: May 16, 2024. [Online]. Available: https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct.

[11] Pal K, Bahrainian SA, Mercurio L, Eickhoff C, 2023, "Neural Summarization of Electronic Health Records", JMIR Preprints, DOI 10.2196/preprints.49544.

Yale at "Discharge Me!": Evaluating Constrained Generation of Discharge Summaries with Unstructured and Structured Information

Vimig Socrates MS[1,3]* Thomas Huang BS[1,2]*
Xuguang Ai MS[1], Soraya Fereydooni BS[1] Qingyu Chen PhD[1]
R. Andrew Taylor MD MHS[1,2] David Chartash PhD[1,4]
[1]Department of Biomedical Informatics and Data Science, Yale University School of Medicine
[2]Department of Emergency Medicine, Yale School of Medicine
[3]Program of Computational Biology and Bioinformatics, Yale University
[4]School of Medicine, University College Dublin - National University of Ireland, Dublin
{vimig.socrates,t.huang}@yale.edu | * = Equal Contribution

Abstract

In this work, we propose our top-ranking (2nd place) pipeline for the generation of discharge summary subsections as a part of the BioNLP 2024 Shared Task 2: "Discharge Me!". We evaluate both encoder-decoder and state-of-the-art decoder-only language models on the generation of two key sections of the discharge summary. To evaluate the ability of NLP methods to further alleviate the documentation burden on physicians, we also design a novel pipeline to generate the brief hospital course directly from structured information found in the EHR. Finally, we evaluate a constrained beam search approach to inject external knowledge about relevant patient problems into the text generation process. We find that a BioBART model fine-tuned on a larger fraction of the data without constrained beam search outperforms all other models.

1 Introduction

Discharge summaries are vital sources of information that provide a bridge between inpatient treatment and continuation of care in rehabilitation, outpatient, or other intermediate settings. These summaries are often the only form of communication that follows a patient to their next setting of care (Kind and Smith, 2011). This documentation serves many roles, including next action items necessary for the patient, clear identification of incidental findings necessitating follow-up, new treatment regiments, and many other important components of patient treatment plan (Chatterton et al., 2023). The discharge summary is a complex document that addresses not only a wide array of members of the care team including the patient's primary care physician, specialists, ancillary departments, but also the patient themselves. Within the discharge summary, two sections are particularly instrumental in the continuity of care and complex in their content: the Brief Hospital Course (BHC) and the Discharge Instructions.

The BHC summarizes the course of events that occurred from the moment a patient presents to the emergency department (ED) through their hospital course, ending in discharge. This summary is often structured by problem list or procedure and depends heavily on the discharging service (medical vs. surgical etc.) Discharge instructions serve to inform the patient through lay language about key details of their hospital stay, as well as to structure the complex follow-up care that patients will navigate after discharge, enabling them to manage their health effectively in collaboration with their outpatient medical team (Becker et al., 2021; Dubb et al., 2022).

Large Language Models (LLMs), such as ChatGPT, offer a potential solution to the long-standing issue of inaccessible medical communication and the time-demanding nature of synthesis of discharge summaries. Creating high-quality discharge summaries is a challenging and time-consuming task. Significant prior work has demonstrated the broad capabilities of LLMs in clinical natural language processing (Gilson et al., 2023; Nayak et al., 2023; Eppler et al., 2023; Zaretsky et al., 2024). This work suggests that LLMs could be leveraged for the automated generation of discharge summaries. The automatic generation of discharge summaries from inpatient documentation could support alleviating the burden of clinical documentation, particularly under the significant pressures of the inpatient setting (Searle et al., 2023; O'Donnell et al., 2009).

2 Background

2.1 Task Description

The BioNLP 2024 Task 2 Challenge, *Discharge Me!* (Xu et al., 2024), consists of two subtasks: (1) generation of Brief Hospital Courses (BHC) and (2) generation of Discharge Instructions.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 724–730
August 16, 2024. ©2024 Association for Computational Linguistics

Task	Model	Overall	BLEU	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	Meteor	AlignScore	MEDCON
Brief Hospital Course	AIMI-Baseline	0.1141	0.0171	0.1184	0.0698	0.1348	0.1726	0.0889	0.1714	0.1398
	GPT 3.5 (0-shot)	0.2035	0.0210	0.3472	0.0983	0.2289	0.2815	0.2232	0.1865	0.2410
	Clinical-T5	0.2068	0.0357	0.3145	0.1378	0.2273	0.3180	0.1678	**0.2251**	0.2285
	BioBART	0.2198	0.0576	0.3161	0.1100	0.2021	**0.3383**	**0.2823**	0.2007	**0.2515**
	BioBART v2	0.2227	**0.0600**	0.3310	0.1239	0.2231	0.3354	0.2802	0.1941	0.2340
	BioBART-Shuffled	**0.2464**	0.0488	**0.3807**	**0.2052**	**0.3003**	0.3278	0.2661	0.1959	0.2463
	GPT-3.5 + Pseudo-SOAP notes*	0.1498	0.0032	0.2603	0.0345	0.1233	0.2374	0.2037	0.2000	0.1360
	BioBART + Constrained Generation	0.1255	0.0045	0.2015	0.0381	0.0900	0.1607	0.2070	0.1739	0.1282
Discharge Instructions	AIMI-Baseline	0.0909	0.0119	0.1343	0.0335	0.0910	0.1026	0.0889	0.1622	0.1025
	GPT-3.5 (0-shot)	0.2289	0.0299	0.3761	0.1312	0.2271	0.3047	0.3109	0.1821	0.2690
	BioBART	**0.3308**	**0.1458**	**0.4465**	**0.1796**	**0.2679**	**0.4382**	**0.3976**	**0.3527**	**0.4183**

Table 1: Results for Black-box GPT models, BART, and T5 pipelines for brief hospital course (subtask 1) and discharge instruction (subtask 2) generation. * indicates that only 121/250 summaries were used, as not all patients had transfer events in structured data.

Formally, we define both problems as sequence-to-sequence text generation tasks. Subtask 1 can be seen as abstractive summarization of the text preceding the BHC. As much of this text is auto-generated by the EHR (e.g. demographics, past medical history, pertinent labs), we can leverage this information without increasing the burden of documentation on physicians. Subtask 2 can also be considered summarization, but requires that the generated text be patient-readable. In this setting, we use the BHC that would have already been generated and attempt to simplify the hospital course, while also providing recommendations for follow-up care. In this work, we evaluate both encoder-decoder models (e.g. BART, T5) and decoder-only models (black-box Azure GPT-3.5). We also propose 2 additional pipelines: (1) a structured data-only BHC generation pipeline that completely removes the need for physician documentation and (2) a constrained beam search approach to improve recall of clinical concepts in BHCs.

2.2 Dataset

The challenge dataset included discharge summaries from 109,168 visits to the Emergency Department (ED) from the **note** and **ED** modules of MIMIC-IV. MIMIC-IV is a publicly available database sourced from the Beth Israel Deaconess Medical Center electronic health record (EHR) that provides a wide array of de-identified patient information containing both structured and unstructured data (Johnson et al., 2023). The text data consisted of a discharge summary, chief complaints, and at least one radiology report. The dataset also included demographics and ED diagnoses as structured data. For our models developed using only structured information, we used data from the MIMIC-IV **hosp** module that included additional demographics (e.g. admission times, treatment wards), hospital diagnoses, procedures, laboratory values, inpatient medications, and lab culture results. These structured data elements were used in a GPT-3.5 pipeline described in Section 3.2.3. The data set was divided into training, validation, and testing (phase I and II) testing sets, of which 250 discharge summaries were selected for standardized final evaluation (Xu et al., 2024).

2.3 SOAP Notes

The subjective, objective, assessment, and plan (SOAP) note is a widely used standard method of documentation used by healthcare providers. The SOAP note is a method of standardizing medical documentation to help physicians streamline clinical decision making (Weed, 1968). The subjective commonly contains the chief complaint, history of present illness (HPI), past medical history, and review of systems. Objective information contains vital signs, physical exam findings, and diagnostic data such as labs, imaging, and other testing. The assessment represents a synthesis of the information collected in prior sections and a presentation of a differential diagnosis. The plan reflects the next steps, frequently including important action items such as consults, additional testing, medications, and other interventions (Tait, 1979).

3 Method

3.1 Data Preprocessing

For the generation of BHCs, we extract all preceding text prior to the brief hospital course. For the generation of discharge instructions, we use the provided ground-truth BHC. For all models that we fine-tuned, we tokenized text based on the encoding scheme used during model training. BART-based models (Lewis et al., 2020) use Byte-pair encoding (Radford et al., 2019) and truncate input and output tokens to the max sequence length of 1024.

Unlike BART-based models, T5 models (Raffel et al., 2020) use Sentencepiece tokenization (Kudo and Richardson, 2018) and relative position embeddings, so while input tokens are truncated to 512 (max context length), output tokens are set to the maximum for our dataset (3903 tokens).

3.2 Subtask 1: Brief Hospital Course

3.2.1 Generation from Unstructured Data

We train three model classes to generate brief hospital courses: BART-based, T5-based, and black box GPT models. We opt for continuously-pretrained biomedical encoder-decoder models as previous work has demonstrated that these models outperform those trained from scratch (Gu et al., 2021). *BioBART-Large* (Yuan et al., 2022) is a 12-layer encoder-decoder model with 406M parameters initialized from a general domain model and continuously pretrained on biomedical paper abstracts from PubMed. The other encoder-decoder model, *Clinical-T5-Large* (Hernandez et al., 2023), with 770M parameters, was instead trained from scratch on MIMIC-III (Johnson et al., 2016) and MIMIC-IV notes. Due to the relative position embeddings in Clinical-T5, it can generate longer summaries, unlike the BioBART model, which is limited to 1024 tokens of output, due to its fixed position embeddings. Given that 10.3% of the challenge dataset has greater than 1024 tokens, we hypothesize that the Clinical-T5 model will achieve better performance.

We also compare fine-tuning with 0-shot performance of an Azure OpenAI GPT-3.5 Turbo model[*] with human-based abuse monitoring switched off, in keeping with MIMIC's data use agreement. During preliminary evaluations, no significant differences were observed between GPT-3.5 and GPT-4, leading us to choose the more economical option.

3.2.2 Constrained Generation

Upon manual review during phase II testing, we noticed that our encoder-decoder models often failed to provide key formatting or content sections in the BHC. For example, summaries generated by CMED (Cardiac Medical) services tend to contain summaries structured by problem list (e.g. *# UTI:...# Cough...*). Due to the variability in discharge summaries based on individual physician preferences, discharge ward, and patient context, encoder-decoder models seemed to struggle to learn summary structure. Therefore, we attempted to enforce the inclusion of important problems through constrained beam search generation (Anderson et al., 2016; Post and Vilar, 2018; Hu et al., 2019). Constrained beam search injects external knowledge into the generative beam search process by including additional beams for tokens of interest. To identify relevant concepts of interest, we used MedCat (Kraljevic et al., 2021) to tag the history of present illness section preceding the BHC with UMLS (Bodenreider, 2004) concepts. We then constrained our best-performing BioBART model to include these concepts during its beam search. We called this model **BioBART + Constrained Generation**.

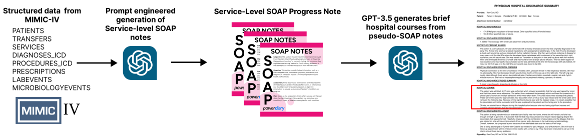

Structured data from MIMIC-IV
PATIENTS
TRANSFERS
SERVICES
DIAGNOSES_ICD
PROCEDURES_ICD
PRESCRIPTIONS
LABEVENTS
MICROBIOLOGYEVENTS
MIMIC IV
Prompt engineered generation of Service-level SOAP notes
Service-Level SOAP Progress Note
SOAP NOTES
SOAP
powerdiary
GPT-3.5 generates brief hospital courses from pseudo-SOAP notes
PHYSICIAN HOSPITAL DISCHARGE SUMMARY
HOSPITAL COURSE

Figure 1: GPT-3.5 Pipeline for generation of brief hospital courses using only structured data

3.2.3 Generation from Structured Data

To evaluate the ability of GPT-based models to further alleviate documentation burden, we develop a pipeline to generate BHCs directly from structured data. As shown in Figure 1, we first extract all relevant structured information for each patient: demographics, ED diagnoses, procedures, inpatient medications, lab values, and lab culture results. As SOAP notes are generally generated either daily or for each service, we generate individual SOAP notes for each transfer during the patient's hospital admission. These SOAP notes are then provided to the GPT-3.5 model in a 0-shot setting to generate brief hospital courses.

3.3 Subtask 2: Discharge Instructions

Similar to subtask 1, we evaluate both fine-tuning and in-context learning (ICL) in the generation of discharge instructions. Namely, we fine-tune BioBART-Large on the brief hospital course text, under the assumption that discharge summaries are generated sequentially and this information would be available to the model in practice. A limited subset of BHCs was provided to GPT-3.5 LLM in a 2-shot approach. In this setting, we noticed that in-context learning did not necessarily improve generation structure so we opted to not evaluate the full test set in the 2-shot setting. Therefore, we

[*] version: 2023-07-01-preview

provided GPT-3.5 with BHCs and evaluated it in a 0-shot setting. GPT-4 LLM also did not demonstrate performance improvement as measured by ROUGE in a limited subset of 375 notes (GPT-3.5 R-1: 0.306, R-2: 0.083, R-L: 0.159, GPT-4 R-1: 0.309, R-2: 0.079, R-L: 0.157, respectively). As a result, GPT-3.5 was provided the entire test set of BHCs in a 0-shot setting for discharge instruction generation.

4 Experiments and Results

4.1 Quantitative Evaluation

To evaluate the performance of the models, a suite of automated summarization metrics including BLEU, ROUGE-1, ROUGE-2, ROUGE-L, BERTScore, Meteor, AlignScore, and MEDCON were calculated (Papineni et al., 2002; Lin, 2004; Zhang et al., 2019; Banerjee and Lavie, 2005; Zha et al., 2023; Yim et al., 2023). We report summarization metrics for all model variations in Table 1. On BHC generation, we trained two encoder-decoder models: the *Clinical-T5* and *BioBART* models finding that BioBART performed better. Manual review showed that while Clinical-T5 was fine-tuned on larger generations (up to 3903 tokens), its original pre-training truncated generation to 512 tokens, and therefore the model remained biased towards shorter generations. To evaluate the impact of in-domain vocabulary on BHC generation, we also tested *BioBART v2*, continuously pre-trained with a larger, cross-domain vocabulary, as opposed to the standard general domain vocabulary (Yuan et al., 2022). We found that BioBART outperformed BioBART v2, potentially due to the expanded vocabulary coming from biomedical literature rather than the clinical notes found in this challenge. Finally, we also tested the impact of increased training data size (*BioBART-Shuffled*) by shuffling the phase I training, validation and testing data set, before recombining for an additional 14,690 discharge summaries in the training set. Across the encoder-decoder models, *BioBART-Shuffled* performed best, yielding us 2nd place on the challenge leaderboard.

We also compared these results to a *GPT-3.5 0-shot* model, finding that black-box GPT-3.5 performed worse than the best performing fine-tuned model. When repeating this experiment with our structured data-only method (*GPT-3.5 + Pseudo-SOAP notes*) as well as constrained generation, we found that neither of these methods offered improvement over our best-performing model, BioBART. In the generation of discharge instructions, the BioBART outperformed GPT-3.5, and so was included as our challenge submission.

Evaluation Criteria	Brief Hospital Course	Discharge Instructions
Completeness	3.52	4.27
Correctness	2.57	3.95
Readability	2.11	-
Overall	1.53	2.36

Table 2: Average ratings across 3 criteria for 3 clinicians (Discharge instruction readability was not assessed as the target audience are patients)

4.2 Qualitative Evaluation

To evaluate the clinical validity of generated brief hospital courses and discharge instructions, a team of 3 clinicians reviewed a random sample of 25 generations from the hidden set of 250 discharge summaries. Each clinician rated both the brief hospital courses and discharge instructions according to 3 criteria: Completeness (captures important information), Correctness (contains less false information), and Readability. They also provide a holistic assessment as an overall score (Xu et al., 2024). All metrics are averaged, and results are presented in Table 2, showing that both the brief hospital course and discharge instructions received their highest grades in completeness and lowest in the overall evaluation criteria. Selected discharge instructions received higher grades across completeness, correctness, and overall criteria compared to brief hospital course. This is likely due to the nature of the increased complexity and wider range of information often necessary for inclusion in a brief hospital course.

5 Limitations

While our methods were able to produce reasonable BHCs and discharge instructions, there are several important limitations to our study. Computationally, we were unable to perform a rigorous hyperparameter search across all our experimental conditions due to computational constraints. There is potential for improvement given further resources. Namely, we were surprised that *Constrained Generation* performed significantly worse than vanilla BioBART models. This is potentially due to the additional hyperparameters that need to be tuned, including the tokens of interest that MedCat identified and beam sizes.

Furthermore, we believe that there is limited clinical validity in the current task as it has been framed. The automated generation of BHC and discharge instructions utilizing physician generated preceding text does not truly automate the task, nor does it obviate the need for the core summarization task of note-writing on the part of the physician (rather than documentation of findings). We attempted to model a more representative use case by including the generation of Pseudo-SOAP notes but found significantly worse performance, demonstrating this difficulty of the real-world clinical task. Furthermore, the format and physician- and institution-specific stylistic choices had a significant impact on automated performance, as demonstrated by the significant variation in documentation length and lack of standard templates even within services that discharged patients. The Challenge organizers did attempt to alleviate concerns around generalizability with a qualitative analysis by clinicians, but further efforts in automated metrics involving semantic comparison are necessary.

6 Conclusion

In this work, we present experiments for the automated generation of brief hospital courses and discharge instructions from structured and unstructured data captured during an ED encounter. We find that a BioBART model with increased training data performed better than both other encoder-decoder models and a black-box decoder-only model. We also find that constraining generation to emphasize generation of UMLS concepts worsens performance. Finally, we show that GPT-3.5 can generate brief hospital courses purely from structured information, further reducing the annotation burden for physicians.

Acknowledgments

We would like to thank Justin Xu and the other members of the "Discharge Me!" team for organizing this competition, as well as all their feedback and prompt responses to questions along the way.

References

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Guided open vocabulary image captioning with constrained beam search. *arXiv preprint arXiv:1612.00576*.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Christoph Becker, Samuel Zumbrunn, Katharina Beck, Alessia Vincent, Nina Loretz, Jonas Müller, Simon A Amacher, Rainer Schaefert, and Sabina Hunziker. 2021. Interventions to improve communication at hospital discharge and rates of readmission: a systematic review and meta-analysis. *JAMA Network Open*, 4(8):e2119346–e2119346.

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270.

Brittany Chatterton, Jennifer Chen, Eleanor Bimla Schwarz, and Jennifer Karlin. 2023. Primary care physicians' perspectives on high-quality discharge summaries. *Journal of General Internal Medicine*, pages 1–6.

Simran Dubb, Gurmeet Kaur, Sweta Kumari, Krishna Murti, and Biplab Pal. 2022. Comprehension and compliance with discharge instructions among pediatric caregivers. *Clinical Epidemiology and Global Health*, 17:101137.

Michael B Eppler, Conner Ganjavi, J Everett Knudsen, Ryan J Davis, Oluwatobiloba Ayo-Ajibola, Aditya Desai, Lorenzo Storino Ramacciotti, Andrew Chen, Andre De Castro Abreu, Mihir M Desai, et al. 2023. Bridging the gap between urological research and patient understanding: the role of large language models in automated generation of layperson's summaries. *Urology practice*, 10(5):436–443.

Aidan Gilson, Conrad W Safranek, Thomas Huang, Vimig Socrates, Ling Chi, Richard Andrew Taylor, David Chartash, et al. 2023. How does chatgpt perform on the united states medical licensing examination (usmle)? the implications of large language models for medical education and knowledge assessment. *JMIR Medical Education*, 9(1):e45312.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23.

Evan Hernandez, Diwakar Mahajan, Jonas Wulff, Micah J Smith, Zachary Ziegler, Daniel Nadler, Peter Szolovits, Alistair Johnson, Emily Alsentzer, et al. 2023. Do we still need clinical language models? In *Conference on Health, Inference, and Learning*, pages 578–597. PMLR.

J. Edward Hu, Huda Khayrallah, Ryan Culkin, Patrick Xia, Tongfei Chen, Matt Post, and Benjamin Van Durme. 2019. Improved lexically constrained decoding for translation and monolingual rewriting. In *Proceedings of the 2019 Conference of the North*

American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 839–850, Minneapolis, Minnesota. Association for Computational Linguistics.

Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. 2023. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

Amy JH Kind and Maureen A Smith. 2011. Documentation of mandated discharge summary components in transitions from acute to subacute care.

Zeljko Kraljevic, Thomas Searle, Anthony Shek, Lukasz Roguski, Kawsar Noor, Daniel Bean, Aurelie Mascio, Leilei Zhu, Amos A Folarin, Angus Roberts, Rebecca Bendayan, Mark P Richardson, Robert Stewart, Anoop D Shah, Wai Keong Wong, Zina Ibrahim, James T Teo, and Richard J B Dobson. 2021. Multi-domain clinical natural language processing with MedCAT: The medical concept annotation toolkit. *Artif. Intell. Med.*, 117:102083.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Ashwin Nayak, Matthew S Alkaitis, Kristen Nayak, Margaret Nikolov, Kevin P Weinfurt, and Kevin Schulman. 2023. Comparison of history of present illness summaries generated by a chatbot and senior internal medicine residents. *JAMA Internal Medicine*, 183(9):1026–1027.

Heather C O'Donnell, Rainu Kaushal, Yolanda Barrón, Mark A Callahan, Ronald D Adelman, and Eugenia L Siegler. 2009. Physicians' attitudes towards copy and pasting in electronic note writing. *Journal of general internal medicine*, 24:63–68.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Matt Post and David Vilar. 2018. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. *arXiv preprint arXiv:1804.06609*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Thomas Searle, Zina Ibrahim, James Teo, and Richard JB Dobson. 2023. Discharge summary hospital course summarisation of in patient electronic health record text with clinical concept guided deep pre-trained transformer models. *Journal of Biomedical Informatics*, 141:104358.

Ian Tait. 1979. *The History and Function of Clinical Records*. Md thesis, University of Cambridge.

Lawrence L Weed. 1968. Medical records that guide and teach. *New England Journal of Medicine*, 278(12):593–600.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10(1):586.

Hongyi Yuan, Zheng Yuan, Ruyi Gan, Jiaxing Zhang, Yutao Xie, and Sheng Yu. 2022. BioBART: Pretraining and evaluation of a biomedical generative language model. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 97–109, Dublin, Ireland. Association for Computational Linguistics.

Jonah Zaretsky, Jeong Min Kim, Samuel Baskharoun, Yunan Zhao, Jonathan Austrian, Yindalon Aphinyanaphongs, Ravi Gupta, Saul B Blecker, and Jonah Feldman. 2024. Generative artificial intelligence to transform inpatient discharge summaries to patient-friendly language and format. *JAMA network open*, 7(3):e240357–e240357.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *arXiv preprint arXiv:2305.16739*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

IgnitionInnovators at "Discharge Me!": Chain-of-Thought Instruction Finetuning Large Language Models for Discharge Summaries

An Quang Tang
RMIT University, Australia
s3695273@rmit.edu.vn

Xiuzhen Zhang
RMIT University, Australia
xiuzhen.zhang@rmit.edu.au

Minh Ngoc Dinh
RMIT University, Australia
minh.dinh4@rmit.edu.vn

Abstract

This paper presents our proposed approach to the Discharge Me! shared task, collocated with the 23th Workshop on Biomedical Natural Language Processing (BioNLP). In this work, we develop an LLM-based framework for solving the Discharge Summary Documentation (DSD) task, i.e., generating the two critical target sections 'Brief Hospital Course' and 'Discharge Instructions' in the discharge summary. By streamlining the recent instruction-finetuning process on LLMs, we explore several prompting strategies for optimally adapting LLMs to specific generation task of DSD. Experimental results show that providing a clear output structure, complimented by a set of comprehensive Chain-of-Thoughts (CoT) questions, effectively improves the model's reasoning capability, and thereby, enhancing the structural correctness and faithfulness of clinical information in the generated text. Source code is available at: https://anonymous.4open.science/r/Discharge_LLM-A233

1 Introduction

Discharge summaries encapsulate key details of a patient's hospitalization, from admission to discharge. These documents, however, can contain excessive amount of medical notes, making it difficult for subsequent caregivers or patients to quickly understand essential past medical information. *Brief Hospital Course* and *Discharge Instructions* then become two critical sections in discharge summaries to address this issue. The former outlines critical hospital events for healthcare providers, while the latter offers post-discharge care instructions to patients and their caregivers. The Discharge Me! shared task [1] (Xu et al., 2024) at the BioNLP Workshop, known as Discharge Summary Documentation (DSD), focuses on efficiently generating these two critical sections, a task that is both challenging and time-consuming for clinicians.

In this paper, we introduce a novel LLM-based framework, namely Discharge-LLM, for the DSD task (Xu et al., 2024). Discharge-LLM employs modern prompting strategies (e.g., Chain-of-Thought (CoT)) into instruction-finetuning a `Mistral` Large Language Model (LLM), which enhances structural correctness and faithfulness of clinical information to generate the Brief Hospital Course and Discharge Instructions sections of discharge summaries.

2 Related Work

In recent years, Large Language Models (LLMs) like GPT-2 or GPT-3 have excelled in NLP tasks such as language generation, question answering, due to their vast number of paramters and extensive training on diverse datasets. These models can be adapted to new domains and tasks through methods like prompting, which uses natural language instructions (Liu et al., 2023) or few-shot examples (Lampinen et al., 2022). However, considering DSD problem, the length and excessive information in discharge summaries hinders their use as examples for few-shot prompting. Alternatively, parameter-efficient fine-tuning, which freezes an LLM weights and inserts a small number of tunable parameters (Lin et al., 2020), has proven effective in specialized clinical tasks like radiology report generation (Van Veen et al., 2023).

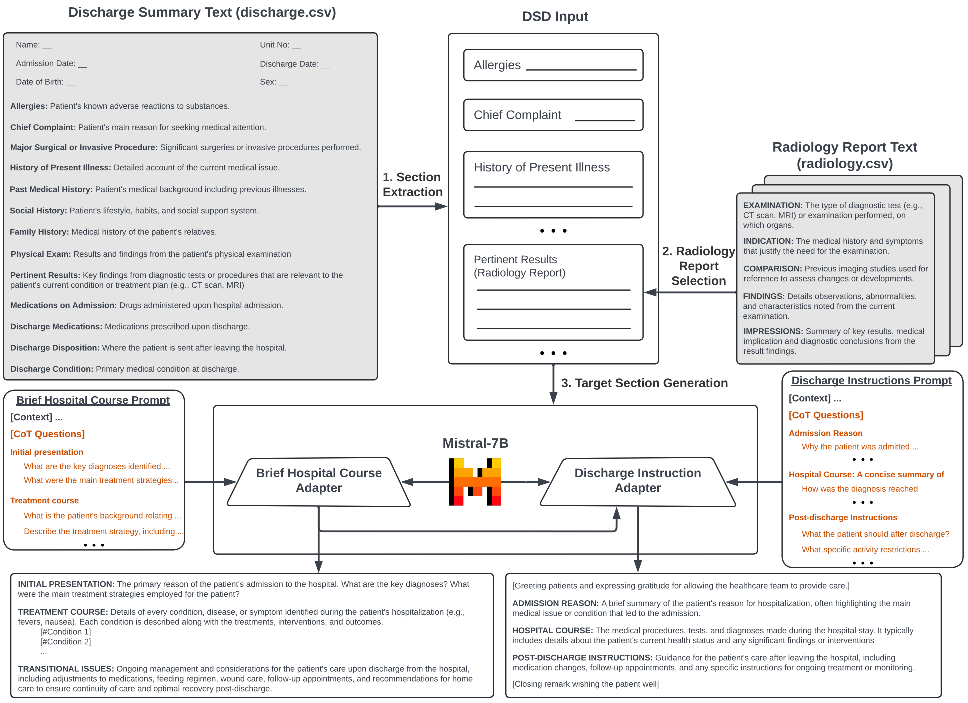
From the clinical summarization perspective, research towards th DSD task was very limited. But there has been a growing focus on many clinical text generation tasks, encompassing radiology reports (Ben Abacha et al., 2021), clinical notes (Grambow et al., 2022), and summary of diagnoses and patient problems (Gao et al., 2022).

3 Problem description

A discharge summary can contain several free-text sections and medical notes compiled from EHR.

[1] https://www.codabench.org/competitions/2008/

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 731–739
August 16, 2024. ©2024 Association for Computational Linguistics


Discharge Summary Text (discharge.csv)
DSD Input
Allergies
Chief Complaint
History of Present Illness
Pertinent Results (Radiology Report)
1. Section Extraction
2. Radiology Report Selection
Radiology Report Text (radiology.csv)
3. Target Section Generation
Mistral-7B
Brief Hospital Course Adapter
Discharge Instruction Adapter
Brief Hospital Course Prompt
Discharge Instructions Prompt

Name: __ Unit No: __
Admission Date: __ Discharge Date: __
Date of Birth: __ Sex: __

Allergies: Patient's known adverse reactions to substances.

Chief Complaint: Patient's main reason for seeking medical attention.

Major Surgical or Invasive Procedure: Significant surgeries or invasive procedures performed.

History of Present Illness: Detailed account of the current medical issue.

Past Medical History: Patient's medical background including previous illnesses.

Social History: Patient's lifestyle, habits, and social support system.

Family History: Medical history of the patient's relatives.

Physical Exam: Results and findings from the patient's physical examination

Pertinent Results: Key findings from diagnostic tests or procedures that are relevant to the patient's current condition or treatment plan (e.g., CT scan, MRI)

Medications on Admission: Drugs administered upon hospital admission.

Discharge Medications: Medications prescribed upon discharge.

Discharge Disposition: Where the patient is sent after leaving the hospital.

Discharge Condition: Primary medical condition at discharge.

EXAMINATION: The type of diagnostic test (e.g., CT scan, MRI) or examination performed, on which organs.

INDICATION: The medical history and symptoms that justify the need for the examination.

COMPARISON: Previous imaging studies used for reference to assess changes or developments.

FINDINGS: Details observations, abnormalities, and characteristics noted from the current examination.

IMPRESSIONS: Summary of key results, medical implication and diagnostic conclusions from the result findings.

[Context] ...
[CoT Questions]
Initial presentation
What are the key diagnoses identified ...
What were the main treatment strategies...
Treatment course
What is the patient's background relating ...
Describe the treatment strategy, including ...
• • •

[Context] ...
[CoT Questions]
Admission Reason
Why the patient was admitted ...
• • •
Hospital Course: A concise summary of
How was the diagnosis reached
• • •
Post-discharge Instructions
What the patient should after discharge?
What specific activity restrictions ...
• • •

INITIAL PRESENTATION: The primary reason of the patient's admission to the hospital. What are the key diagnoses? What were the main treatment strategies employed for the patient?

TREATMENT COURSE: Details of every condition, disease, or symptom identified during the patient's hospitalization (e.g., fevers, nausea). Each condition is described along with the treatments, interventions, and outcomes.
[#Condition 1]
[#Condition 2]
...

TRANSITIONAL ISSUES: Ongoing management and considerations for the patient's care upon discharge from the hospital, including adjustments to medications, feeding regimen, wound care, follow-up appointments, and recommendations for home care to ensure continuity of care and optimal recovery post-discharge.

[Greeting patients and expressing gratitude for allowing the healthcare team to provide care.]

ADMISSION REASON: A brief summary of the patient's reason for hospitalization, often highlighting the main medical issue or condition that led to the admission.

HOSPITAL COURSE: The medical procedures, tests, and diagnoses made during the hospital stay. It typically includes details about the patient's current health status and any significant findings or interventions

POST-DISCHARGE INSTRUCTIONS: Guidance for the patient's care after leaving the hospital, including medication changes, follow-up appointments, and any specific instructions for ongoing treatment or monitoring.

[Closing remark wishing the patient well]

Figure 1: The Discharge-LLM framework

The task is to generate the *Brief Hospital Course* (BHC) and *Discharge Instructions* (DI) sections, leveraging readily available data in other sections of discharge summaries and additional information about a patient's admission (e.g.,radiology, stays) from the dataset. Brief Hospital Course outlines critical hospital events for healthcare providers, while Discharge Instructions offers post-discharge care instructions to patients and their caregivers.

4 Methodology

We propose the Discharge-LLM framework, which adapt LLM to the each generation task of DSD, illustrated in Figure 1. Discharge-LLM applies three steps, namely *Section Extraction*, *Radiology Report Selection* and *Target Section Generation* to generate the two critical target sections given discharge summary and radiology report information of a patient's hospital visit. Note that we utilize the generated BHC for the subsequent generation of DI. Table 4 and 5 (Appendix A) show the output of the two target sections generated by our framework.

4.1 Section Extraction

To generate the two target sections BHC and DI, the most straightforward approach is to leverage the other readily available free-text sections in the discharge summary as the input for the generation stage. But using them all for one-stage generation is overwhelming and prone to hallucination because some sections are irrelevant or contain thousand words of nonessential information, making key aspects of the patient's record often be omitted. We thereby design sets of heuristics (e.g., regular expressions) to selectively extract clinical notes information from 13 relevant sections of the discharge summaries (e.g, *History of Present Illness*, *Pertinent Results*, . . .), with definition of each section described in Figure 1. We report data distribution of these sections in Table 6 (Appendix B)

4.2 Radiology Report Selection

Through exploring the format of different sections of the discharge summary, we notice a great complications in the structure and content of the *Pertinent Results* section, likely due to note bloat and information overload. This section, intended to highlight key findings of radiologist to the patient's treatment, is often cluttered with excessive laboratory and imaging data (e.g., blood tests, CT scans). These extraneous details can lead to challenges such as hallucination and high resource demands in generative tasks. Consequently, we explored using radiology reports as a viable alternative. These reports, often duplicated partially or entirely in the Pertinent Results section, succinctly convey diagnoses corresponding to specific lab results. We selected radiology reports with similar Impressions to those in the Pertinent Results and used these as a substitute, streamlining the content effectively.

4.3 Target Section Generation

In this framework, we performed instruction-finetuning on LLM to adapt the model to DSD. For computational feasiblity, we employed Low-Rank Adaptation (LoRA) (Hu et al., 2021), a parameter-efficient fine-tuning method that adds a small number of trainable parameters to the model while freezing the model's original weight, resulting in standalone adapters. The adapters, specifically fine-tuned for each generation task in DSD, adjust important weight of LLMs to capture and generate clinical information in the corresponding form.

Prompting Strategies Following OpenAI's prompt engineering guidelines [2], we structured our prompts into five parts, detailed in Table 7 (Appendix D): 1) Context of the discharge summary input to be summarized 2) Definition of the generation task and the specific section for documenting the discharge summary 3) Structure of the expected output of the generating section, infused with 4) Set of Chain-of-Thought (CoT) questions expected to be answered by the LLMs to capture and generate the information in each subsection of the output. Of those, our primary strategy is Part 5, which involved curating effective and generalizable CoT questions based on analysis of numerous samples. This manual effort helped in designing templates and questions that effectively guide the LLMs to focus on critical information amidst the extensive data and noise in the discharge summaries. We analyzed the medical questionnaire essential for each section, based on hundreds of samples, in Appendix C , which underpins our CoT questions and prompt design.

5 Experiments

5.1 Baseline and Implementation Details

To showcase the utility of prompt designing for adaptation to the DSD task, we developed three baselines, corresponding to three prompt variants for instruction-finetuning LLMs. $\text{Discharge\_LLM}_{\text{Base}}$ was fine-tuned with no instruction, but only the discharge summaries as input and the respective target section as output. $\text{Discharge\_LLM}_{\text{Context}}$ was fine-tuned with additional natural langauge instructions as prefix to the discharge summary to provide the context and definition of the task's input/output. Finally, $\text{Discharge\_LLM}_{\text{CoT}}$ was fine-tuned using prompts outlining the structure of the respective generating target section. Along the structure, we embed some CoT questions to elicit LLMs to generate output aligned with the questions.

We choose `Mistral` [3] (Jiang et al., 2023) as our LM. The LLM was fine-tuned on a NVIDIA RTX 4090 GPU, and took 10 hours for fine-tuning each generation task. The following hyperparameters were used: 1 sample per device, a LoRA rank and alpha of 128 and 64 for parameter-efficient fine-tuning, a learning rate of $2e10 - 4$. We keep other hyperparameters to their default values.

Metrics We followed the organizers to measure textual similarity and factual correctness of the generated text based on several metrics, including BLEU-4 (Papineni et al., 2002), ROUGE (Lin, 2004), BERTScore (Zhang et al., 2019), Meteor (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023).

Dataset The dataset for this task was sourced from the MIMIC-IV (Johnson et al., 2023) dataset, including 109,168 emergency department (ED) admissions and were split into a training (68,785), a validation (14,719), a phase I testing (14,702), and a phase II testing (10,962) subsets.

Data Preprocessing To address the variation in discharge summary length, we select data within the interquartile range (Q1-Q3) for training and validation. We further ensure consistency by selecting only samples with discharge summaries containing all 13 common sections and their target sections follow the most common format, as outlined in Figure 1. Overall, 11.1k and 8.7k samples were selected for training of BHC and DI generation, respectively. For experiment, due to runtime and computational limitations, we sample 250 hidden entries from each phase's testing data, totaling 500 samples for evaluation of each generation task.

5.2 Results

Table 1 presents the performance of models fine-tuned by different prompt variants. Overall, in both generation tasks, natural language instructions plays a critical role in guiding the LLM with comprehensive knowledge to understand the task. Providing well-described context of the generation task already helps the model achieves up to 14% of

[2] `https://platform.openai.com/docs/guides/prompt-engineering`

[3] https://mistral.ai/

Framework	R-1	R-2	R-L	BLEU	BERTScore	Meteor	AlignScore	MEDCON
Discharge_LLM$_{\text{CoT}}$	**0.283**	0.087	0.170	**0.062**	**0.368**	**0.206**	0.230	**0.408**
Discharge_LLM$_{\text{Context}}$	0.263	**0.091**	**0.178**	0.058	0.365	0.191	**0.234**	0.397
Discharge_LLM$_{\text{Base}}$	0.240	0.074	0.159	0.043	0.347	0.170	0.221	0.376

(a) Brief Hospital Course Generation

Framework	R-1	R-2	R-L	BLEU	BERTScore	Meteor	AlignScore	MEDCON
Discharge_LLM$_{\text{CoT}}$	**0.392**	**0.151**	**0.246**	**0.077**	**0.373**	**0.272**	**0.288**	**0.452**
Discharge_LLM$_{\text{Context}}$	0.356	0.103	0.205	0.075	0.360	0.272	0.286	0.429
Discharge_LLM$_{\text{Base}}$	0.335	0.102	0.215	0.041	0.324	0.181	0.251	0.318

(b) Discharge Instructions Generation

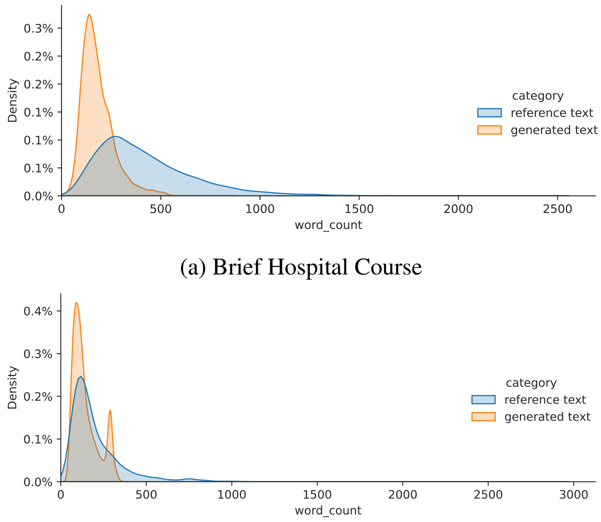
Table 1: Evaluation of prompt variants for finetuning Discharge-LLM

Framework	R-1	R-2	R-L	BLEU	BERTScore	Meteor	AlignScore	MEDCON	**Overall**
Discharge_LLM$_{\text{CoT}}$	0.370	0.131	0.245	0.068	0.360	0.314	0.215	0.324	**0.253**
Best ranked system	0.453	0.201	0.308	0.124	0.438	0.403	0.315	0.411	**0.332**

Table 2: Overall performance on two target section from the shared task's phase 2 leaderboard

Target Section	*min*	*median*	*mean*	*max*
BHC	22	367	425	2439
DI	10	153	201	2900

Table 3: Statistics of reference text's word count on phase 2's test set



(a) Brief Hospital Course

(b) Discharge Instructions

Figure 2: Distribution of samples per number of words on phase 2's test set

performance gain across the metrics and tasks. Further, infusing CoT questions into the instructions effectively elicit LLM to think better, providing an additional 9% performance increase. Notably, a reasonable improvement on MEDCON score also indicates better accuracy and consistency of clinical concepts in the generated text.

5.3 Shared Task's Evaluation Results

Table 2 summarizes our framework's overall performance on the phase 2 test set of the shared task, alongside the best-ranked system [4]. We notice there is still a gap between our Discharge_LLM$_{\text{CoT}}$ framework and the best ranked system, of which the Overall score is 0.332. This performance dip is common across submissions, likely due to prevalent data quality issues in the Discharge Summary Documentation (DSD) task. DSD, a real-world summarization challenge, involves processing actual discharge summary information with significant variability in formatting and length. Figure 2 shows word distribution variances in the target sections. It is noticeable our models are trying to set a common length for the target sections, and are struggling to converge to the wide range of lengths of the reference text, highlighted by Table 3. We note slight performance variation of Discharge_LLM$_{\text{CoT}}$ in terms of Meteor, AlignScore and MEDCON, in between Table 1 and 2. A possible reason for such variation may be that hidden test data in the shared task has distributions significantly deviate from the publicly released data for model development. Furthermore, due to computational constraints and a focus on high-quality data, we utilized only a subset of the available training data. With more comprehensive training, we anticipate improved model convergence.

6 Conclusion

In this paper, we present a LLM-based framework for Discharge Summary Documentation that adopts several prompting strategies into instruction-finetuning an LLM, which enhances structural correctness and faithfulness of clinical information in generated target sections. Using small and open-source LLMs, our work also shows the feasiblity of developing and deploying future lightweight NLP systems locally for confidential clinical tasks.

[4] https://www.codabench.org/competitions/2008/

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Asma Ben Abacha, Yassine Mrabet, Yuhao Zhang, Chaitanya Shivade, Curtis Langlotz, and Dina Demner-Fushman. 2021. Overview of the MEDIQA 2021 shared task on summarization in the medical domain. In *Proceedings of the 20th Workshop on Biomedical Language Processing*, pages 74–85, Online. Association for Computational Linguistics.

Yanjun Gao, Dmitriy Dligach, Timothy Miller, Dongfang Xu, Matthew MM Churpek, and Majid Afshar. 2022. Summarizing patients' problems from hospital progress notes using pre-trained sequence-to-sequence models. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2979–2991.

Colin Grambow, Longxiang Zhang, and Thomas Schaaf. 2022. In-domain pre-training improves clinical note generation from doctor-patient conversations. In *Proceedings of the First Workshop on Natural Language Generation in Healthcare*, pages 9–22, Waterville, Maine, USA and virtual meeting. Association for Computational Linguistics.

Edward Hu, Yelong Shen, Phil Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. 2023. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1.

Andrew Lampinen, Ishita Dasgupta, Stephanie Chan, Kory Mathewson, Mh Tessler, Antonia Creswell, James McClelland, Jane Wang, and Felix Hill. 2022. Can language models learn from explanations in context? In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 537–563, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Zhaojiang Lin, Andrea Madotto, and Pascale Fung. 2020. Exploring versatile generative language model via parameter-efficient transfer learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 441–459, Online. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9).

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Dave Van Veen, Cara Van Uden, Maayane Attias, Anuj Pareek, Christian Bluethgen, Malgorzata Polacin, Wah Chiu, Jean-Benoit Delbrouck, Juan Zambrano Chaves, Curtis Langlotz, Akshay Chaudhari, and John Pauly. 2023. RadAdapt: Radiology report summarization via lightweight domain adaptation of large language models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 449–460, Toronto, Canada. Association for Computational Linguistics.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the first shared task on clinical text generation: Rrg24 and "discharge me!". In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10(1):586.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

A Generated Output of Brief Hospital Course and Discharge Instructions

This section presents details of Table 4 and 5, which show the output of the two "Brief Hospital Course" and "Discharge Instructions" target sections generated by our framework, taken from medical information of patient with $hadm\_id = 21720538$ from the phase 2's test set.

B Data Distribution of Discharge Summary Sections

Table 6 presents the percentage distribution of common sections in the discharge summary text of the training, validation and testing subsets.

C Questionnaire for Discharge Summary Documentation

C.1 Brief Hospital Course

- *Patient Background and Presenting Complaint*: "What is the patient's background including pre-existing medical conditions, and what symptoms or events led to their current hospital admission?"

- Key Diagnoses and Evaluations: "What are the key diagnoses identified during the hospital stay? For each, how was the diagnosis reached, including any significant tests or evaluations conducted?"

- Treatment and Management Strategies: "What were the main treatment strategies employed for the patient's conditions during their stay? Include medications adjusted, procedures performed, and any therapeutic interventions."

- Complications and Additional Diagnoses: "Were there any complications or additional diagnoses during the hospital stay? How were these addressed and managed?"

- Progress and Monitoring: "How did the patient's condition progress throughout the hospital stay, including any monitoring of symptoms, response to treatments, and adjustments made to the treatment plan?"

- Support and Consultation Services: "Which specialist services or support consultations were involved in the patient's care? How did these consultations impact the patient's treatment plan and recovery?"

- Discharge Planning and Instructions: "What were the conditions and considerations for the patient's discharge? Include the discharge medications, any changes from previous medication regimens, and follow-up care or lifestyle recommendations."

- Follow-Up and Post-Discharge Care: "What are the specific follow-up care instructions and any scheduled tests or consultations? Highlight the importance of follow-up for managing ongoing conditions or monitoring recovery."

C.2 Discharge Instructions

- Initial Assessment and Diagnosis: What led to the patient's admission to the hospital, and what were the initial symptoms? Based on the patient's symptoms, what diagnoses were considered and which was confirmed?

- Treatment and Hospital Stay: What treatments were provided to address the patient's symptoms or condition during the hospital stay? Were any surgeries recommended or performed? If a surgery was recommended but not performed, what were the reasons? What were the outcomes of the treatments or interventions provided?

- Patient's Decisions and Care Preferences: Did the patient make any specific requests regarding their care, such as refusing a treatment or requesting a transfer? How were these handled? How did the patient's decisions affect their treatment plan and discharge process?

- Comprehensive Post-Discharge Instructions: What are the general care instructions for the patient after discharge, including diet, activity level, and medication management? Are there any specific symptoms or signs that the patient should monitor for which would require immediate medical attention? How should the patient manage their regular home medications in addition to any new medications prescribed at discharge?

- Activity and Lifestyle Recommendations: What specific activity restrictions or recommendations are given to ensure a smooth recovery? (e.g., weight lifting limits, mobility advice) Are there any restrictions on driving or

```
___ with PMH of NASH vs cryptogenic cirrhosis, HTN,
hyperlipidemia, hypothyroidism who presents from ___
___ with DVT and renal failure.

# DVT: Patient with extensive RLE DVT extending from the right
calf veins into the popliteal vein and femoral vein. Patient
was started on heparin gtt. He was evaluated by ___ who
recommended discharge home with home ___.

# Renal Failure: Patient presented with Cr 8.9, likely
pre-renal in etiology. Patient was given albumin and
diuresis. Cr improved to 3.5 on discharge.

# Cirrhosis: Patient with cirrhosis secondary to NASH vs
cryptogenic cirrhosis. Patient was continued on lactulose and
rifaximin. He was also continued on nadolol.

# Hepatic Encephalopathy: Patient with history of hepatic
encephalopathy. Patient was continued on lactulose and
rifaximin.

# Ascites: Patient with large volume ascites. Patient was
continued on diuresis.

# HTN: Patient was continued on home nadolol.

# Hypothyroidism: Patient was continued on home levothyroxine.

# GERD: Patient was continued on home omeprazole.

# Osteopenia: Patient was continued on home calcium and vitamin
D.

# Allergic Rhinitis: Patient was continued on home cetirizine.

# Chronic Kidney Disease: Patient with chronic kidney disease
secondary to cirrhosis. Patient was continued on home
albumin/dietary management.

# Code: DNR/DNI
# Emergency Contact: ___ (daughter) ___

TRANSITIONAL ISSUES:
- Patient will need to follow up with hepatology and ___
clinic.
- Patient will need to follow up with ___ clinic for
continued management of his DVT.
- Patient will need to follow up with ___ clinic for
continued management of his ascites.
- Patient will need to follow up with ___ clinic for
continued management of his hepatic encephalopathy.
- Patient will need to follow up with ___ clinic for
continued management of his cirrhosis.
- Patient will need to follow up with ___ clinic for
continued management of his hypertension.
- Patient will need to follow up with ___ clinic for
continued management of his hyperlipidemia.
```

Table 4: Generated Output of Brief Hospital Course.

operating machinery, especially if the patient is taking new or continued pain medication?

- Follow-up Care and Monitoring: What follow-up appointments or tests are recommended for the patient? With whom should these appointments be made? How should the patient approach symptom management, especially if they experience pain, dehydration, or other concerning symptoms?

- Communication with Healthcare Providers:

```
Dear Mr. ___,

It was a pleasure taking care of you at ___
___. You were admitted to the hospital because you
were having kidney failure. You were treated with medications
to help your kidneys recover. You also had a blood clot in your
legs. You were started on blood thinners to help prevent
further clots. You were also found to have a lot of fluid in your
belly. You were given medications to help remove the fluid.

You were also found to have a lot of confusion. You were given
medications to help with this. You were also found to have a
blood infection. You were treated with antibiotics. You were
discharged home with hospice care.

We wish you the best.

Sincerely,
Your ___ Team
```

Table 5: Generated Output of Discharge Instructions.

Section	**train**	**valid**	**test (phase 1)**	**test (phase 2)**
Allergies	0.999941669	0.999795571	0.999864232	0.999453801
Chief Complaint	0.999956252	0.999863714	1	0.9997269
Major Surgical or Invasive Procedure	0.518607636	0.516456559	0.517615912	0.517979062
History of Present Illness	0.980386152	0.982010221	0.980788813	0.97997269
Past Medical History	0.960203576	0.96197615	0.958387075	0.960491579
Social History	0.97414472	0.976149915	0.974204059	0.973782431
Family History	0.967567883	0.968245315	0.968094495	0.966317706
Physical Exam	0.978315397	0.980102215	0.978141335	0.977878926
Pertinent Results	0.981231954	0.98153322	0.981942842	0.981429222
Brief Hospital Course	1	1	1	1
Medications on Admission	0.939787675	0.939625213	0.937750322	0.939007738
Discharge Medications	0.980590311	0.981192504	0.980585161	0.981975421
Discharge Disposition	0.989121241	0.987189097	0.987984522	0.987528448
Discharge Diagnosis	0.991950302	0.992231687	0.992261218	0.993172508
Discharge Condition	0.999970834	0.999931857	1	1
Discharge Instructions	1	1	1	1

Table 6: Data Distribution of sections in the discharge summaries in the provided dataset

Under what circumstances should the patient immediately contact their healthcare provider or seek emergency care? What is the recommended way for the patient to communicate with their healthcare team (e.g., phone call, hospital return)?

- Encouragement and Support: How can we encourage the patient to adhere to their discharge instructions and reassure them about their recovery process? What resources or support systems can we recommend to the patient for additional help or information post-discharge?

D Prompts for Discharge Summary Documentation

We present the prompts for generation of the two critical "Brief Hospital Course" and "Discharge Instructions" target sections in Table 7

Prompt for Brief Hospital Course Generation	Prompt for Discharge Instructions Generation
In this task, you are provided with a Discharge Summary delimited by triple quotes. Discharge Summaries are documents that outline the care a patient received during their hospital stay, including diagnoses, treatments, and follow−up care instructions, prepared at the time of a patient's discharge. Discharge Summaries are split into various sections and written under a variety of headings, relating to admission, diagnosis and relevant discharge information. But the provided Discharge summary will be missing the \"Brief Hospital Course\". \"Brief Hospital Course\" is a section of the discharge summaries that outlines the key events of a patient's hospital stay, including the progression from admission to discharge. It is written for the subsequent care providers about the critical aspects of the patient. You are tasked to generate the missing \"Brief Hospital Course\" section in the discharge summary, based on the information of other sections in the discharge summary. Brief Hospital Course outlines the key events of a patient's hospital stay, including the progression from admission to discharge. It is written for the subsequent care providers about the critical aspects of the patient The summary should be written in the following structure, by answering some important questions: 1. Initial presentation: Describe the patient's initial presentation, including the main complaint and relevant history. ∗ What were the main treatment strategies employed for the patient's conditions during their stay? Include medications adjusted, procedures performed, and any therapeutic interventions. ∗ What are the key diagnoses identified during the hospital stay? 2. Treatment course: − For each section header named by "#Condition Name", provide a detailed description of each condition, disease, or symptom of the patient by answering the following questions: ∗ What is the patient's background relating to the condition, disease, or symptom ∗ Describe the treatment strategy, including any medications given, procedures performed, and dietary adjustments. ∗ How was the diagnosis reached, including any significant tests or evaluations conducted? ∗ What were the significant medical or surgical interventions during the hospital stay, including any procedures, diagnostic tests (e.g., CT Scan, Imaging, Blood Test, MRI), and changes in medication? ∗ Were there any complications or additional diagnoses during the hospital stay? How were these addressed and managed? ∗ How did the patient's condition progress throughout the hospital stay, including any monitoring of symptoms, response to treatments, and adjustments made to the treatment plan? ∗ What were the conditions and considerations for the patient discharge? Include the discharge medications, any changes from previous medication regimens, and follow−up care or lifestyle recommendations. 3. Transitional issues: Highlight any transitional care issues addressed during the hospital stay, including changes in medication, dietary adjustments, and specific care instructions. 4. Acute/active issues: Detail the management of acute or active issues encountered during the stay, using the provided structure for each condition. 5. Chronic/stable issues: Summarize how chronic conditions were managed during the stay and any adjustments made to long−term management plans.	In this task, you are provided with a Discharge Summary delimited by triple quotes. Discharge Summaries are documents that outline the care a patient received during their hospital stay, including diagnoses, treatments, and follow−up care instructions, prepared at the time of a patient's discharge. Discharge Summaries are split into various sections and written under a variety of headings, relating to admission, diagnosis and relevant discharge information. But the provided Discharge summary will be missing the \"Discharge Instructions\". \"Discharge Instructions\" is a section of the discharge summaries that summarizes key events of a patient's hospital stay, including the progression from admission to discharge. and provide detailed guidelines to patients (and often their caregivers) upon discharge from a hospital or healthcare facility, outlining how to care for themselves at home. You are tasked to generate the missing \"Discharge Instructions\" section in the discharge summary, based on the information of other sections in the discharge summary. Discharge Instructions summarizes key events of a patient's hospital stay, including the progression from admission to discharge. and provide detailed guidelines to patients (and often their caregivers) upon discharge from a hospital or healthcare facility, outlining how to care for themselves at home. The summary should be written in the following structure, by answering some important questions: 1. Admission Reason: A concise explanation of: ∗ Why the patient was admitted, including any specific conditions or symptoms addressed during the stay. ∗ What was the patient's diagnosis upon admission to the hospital? 2. Hospital Course: A concise summary of what happened to the patient's at the hospital ∗ How was the diagnosis reached, including any significant tests or evaluations conducted (e.g., CT Scan, Imaging, Blood Test, MRI)? ∗ What were the significant medical or surgical interventions during the hospital stay , including any procedures, diagnostic tests (e.g., CT Scan, Imaging, Blood Test, MRI), and changes in medication? ∗ Describe the treatment strategy, including any medications given, procedures performed, and dietary adjustments. ∗ How did the patient respond to the treatment and procedures? Did the patient make any specific requests regarding their care, such as refusing a treatment or requesting a transfer? How were these handled? ∗ Were there any complications or notable improvements in the patient's condition during the stay? ∗ What were the outcomes of the treatments or interventions provided? 3. Post−Discharge Instructions: + Follow−Up Care: ∗ What the patient should do after leaving the hospital? ∗ What specific activity restrictions or recommendations are given to ensure a smooth recovery? (e.g., weight lifting limits, mobility advice) ∗ Are there any restrictions on driving or operating machinery, especially if the patient is taking new or continued pain medication? ∗ Instructions on how to continue treatments started in the hospital, such as new medications or therapy. + Medications (Optional): ∗ Comprehensive instructions for all prescribed medications, including dosage, timing, and any specific instructions for use. ∗ How should the patient manage their regular home medications in addition to any new medications prescribed at discharge? + Monitoring: ∗ Guidelines on any self−monitoring the patient should perform at home, such as weighing themselves, monitoring blood pressure, or blood sugar levels, with instructions on when to contact their healthcare provider. ∗ Are there any specific symptoms or signs that the patient should monitor for which would require immediate medical attention? Under what circumstances should the patient immediately contact their healthcare provider or seek emergency care?

Table 7: Prompts for “Brief Hospital Course” and “Discharge Instructions” generation of the Discharge-LLM framework.

MLBMIKABR at "Discharge Me!": Concept Based Clinical Text Description Generation

Abir Naskar
Computing and Information Systems
University of Melbourne
Parkville VIC 3052
AUSTRALIA
anaskar@student.unimelb.edu.au

Jane Hocking
Population and Global Health
University of Melbourne
Parkville VIC 3052
AUSTRALIA
j.hocking@unimelb.edu.au

Patty Chondros
General Practice and Primary Care
University of Melbourne
Parkville VIC 3052
AUSTRALIA
p.chondros@unimelb.edu.au

Douglas Boyle
General Practice and Primary Care
University of Melbourne
Parkville VIC 3052
AUSTRALIA
dboyle@unimelb.edu.au

Mike Conway
Computing and Information Systems
University of Melbourne
Parkville VIC 3052
AUSTRALIA
mike.conway@unimelb.edu.au

Abstract

This paper presents a method called Concept Based Description Generation, aimed at creating summaries ("Brief Hospital Course" and "Discharge Instructions") using source ("Discharge" and "Radiology") texts. We propose a rule-based approach for segmenting both the source and target texts. In the target text, we not only segment the content but also identify the concept associated with each segment based on text patterns. Our methodology involves creating a combined summarized version of each text segment, extracting important information, and then fine-tuning a Large Language Model (LLM) to generate aspects. Subsequently, we fine-tune a new LLM using a specific aspect, the combined summary, and a list of all aspects to generate detailed descriptions for each task. This approach integrates segmentation, concept identification, summarization, and language modeling to achieve accurate and informative descriptions for medical documentation tasks.

1 Introduction

The "Discharge Me!" (Xu et al., 2024) task within the BioNLP workshop at the Annual Meeting of the Association for Computational Linguistics (ACL) 2024 aims to automate the generation of "Brief Hospital Course" and "Discharge Instructions" sections in discharge notes. These notes are derived from a subset of the MIMIC-IV-Note (Johnson et al., b) and MIMIC-IV-ED (Johnson et al., a) datasets. Hosted on Codabench, the competition provides defined training, validation, and testing sets comprising 109,168 emergency department admissions.

Document statistics from the training data (Table 2 of Appx. A.2) reveal that the average size of the "text" field in Discharge data exceeds 4200 tokens, with over 65,000 documents containing more than 2000 tokens. In this task's dataset, each discharge summary includes a "Brief Hospital Course" section, typically situated after patient history and current treatments, and a "Discharge Instructions" section, commonly found towards the note's conclusion. Evaluation metrics such as BLEU-4 (Papineni et al., 2002), ROUGE-1, ROUGE-2, ROUGE-L (Lin, 2004), BERTScore (Zhang et al., 2020), Meteor (Banerjee and Lavie, 2005), AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023) focus on assessing the textual similarity and factual correctness of the generated text.

Our approach focused on managing source data

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 740–747
August 16, 2024. ©2024 Association for Computational Linguistics

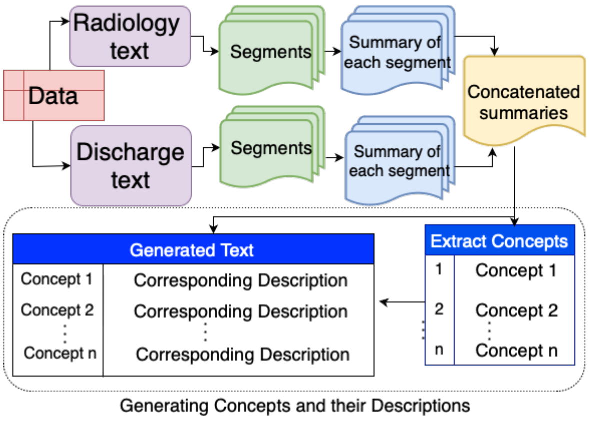


Figure 1: Overview of **Concept Based Description Generation** for generating the *"Brief Hospital Course"* and *"Discharge Instructions"* from Discharge and Radiology text.

size and generating target output systematically to capture crucial information effectively. We aimed to condense lengthy documents while retaining essential details by using a rule-based segmentation method and appropriate prompts for extracting summaries without compromising important information. This strategy allowed us to compress large documents while preserving necessary data for target text generation.

We aim to generate the target text in a structured format by creating summaries that describe specific topics related to the task. These topics are referred to as “concepts”. A concept can encompass various subjects such as patient instructions, medication details, disease information, etc. An example of such a concept is illustrated in Appx. A.1.

Hence, next part of our approach unfolds in two phases. First, we predict the concepts relevant to summarized text. Then, leveraging these concepts, we generate descriptions from the same input text. This process allows us to tailor responses effectively, as the concepts are inherently task-specific, enhancing the accuracy and relevance of our generated content.

2 Related Works

Document summarization, including Query-focused Summarization (QFS), has made significant progress in recent decades. QFS targets specific query information, providing concise answers from retrieved documents. BayeSum by Daumé III and Marcu (2006) leverages multiple documents for state-of-the-art results in query-focused summarization. Vig et al. (2022) explored neural approaches, highlighting their versatility. Baumel et al. (2018) addressed challenges in extractive methods for effective QFS. These efforts showcase diverse strategies in advancing query-focused document summarization.

Varadarajan and Hristidis (2006) introduced query-specific document summarization methods. Additionally, Fu et al. (2020) explored concept extraction in clinical contexts, automatically identifying predefined clinical concepts from unstructured text.

HEPOS by Huang et al. (2021) introduces an innovative encoder-decoder attention mechanism for scalable long document summarization, processing ten times more tokens than traditional models. Moro and Ragazzi (2022) developed the semantic self-segmentation approach to overcome memory limitations in transformer architectures, particularly beneficial in law domains. Grail et al. (2021) proposed a hierarchical propagation layer to enhance reasoning in long document summarization. Pang et al. (2023) suggested a hierarchical inference framework improving summarization models' performance on lengthy texts. Koh et al. (2022) conducted a survey for evaluating research progress and future directions in long document summarization.

Efforts focus on training Large Language Models (LLMs) with medical data, like MIMIC-III (Johnson et al., 2015). Asclepius-R (Kweon et al., 2024) is a specialized clinical large language model trained on synthetic clinical notes created from publicly available case reports, designed to handle patients' clinical notes while addressing privacy and accessibility challenges. BioMistral (Labrak et al., 2024) is tailored for biomedical text, showing superior performance in question-answering and across languages, supporting research in healthcare.

3 Methods

Our pipeline, outlined in Figure 1, consists of four stages and uses three of the six provided data files. The *“text”* fields from the *“radiology”* and *“discharge”* files serve as source documents, while the *“discharge_instructions”* and *“brief_hospital_course”* fields from the *“discharge_target”* data are used for target summaries.

The first stage involves segmenting documents into distinct segments using predefined rules for both source and target data. In the second stage, we use open Large Language Models (LLMs) to generate segment-wise summaries, extracting essential information from each segment of the source text and combining these into a condensed version. For the target texts, we determine the concept for each segment by training LLMs to predict and describe concepts from the summarized text. This process is repeated for both the *"Discharge Instruction"* and *"Brief Hospital Course"* tasks. Finally, the extracted concepts and descriptions are combined and presented as the output for each task.

3.1 Source Document Segmentation

Two datasets serve as the source texts for our analysis: Discharge and Radiology data. A sample snippet from each dataset is illustrated in Figure 2 of Appx. A.1. To segment these documents effectively, we employed distinct strategies tailored to the common patterns found within each dataset. Discharge texts typically contain substantial content and exhibit various types of noise, such as multiple spaces, spaces between new lines, and multiple equal signs used as dividers. Our initial step involves noise removal from both text types. For the Discharge data, we segment the text sections using three consecutive new line characters ("\n\n\n"). Subsequently, we examine specific characters—such as colon (":"), double star ("**"), hash ("#"), and dash ("-")—present in the first line of each segment. If these characters are absent, we merge the segment with its preceding one.

Similar segmentation processes are applied to the Radiology texts, albeit with slight modifications. Here, we split the documents using double new line characters ("\n\n") and search solely for the colon (":") character in the first line of each segment. Segments lacking this character are merged with the previous segment.

3.2 Target Document Segmentation and extracting Concepts

Segmenting target document texts involves both dividing the text and identifying concepts for each segment. Figure 3 of Appx. A.1 shows segments paired with their corresponding concepts, which describe the segment's content. This pattern, though not universal, is common in many documents. Before segmentation, we reduce noise by removing multiple spaces, single spaces, or periods between new lines, and replacing multiple equal signs with a new line character.

Subsequently, we split the entire text of both types using two newline characters ("\n\n"). For the "discharge_instructions" text, we identify common keywords like "Activity", "Medications", "What was done?", "Why was I admitted to the hospital?" etc. as target concepts for generated text. Segments are retained if the first line is in all capital letters or if a colon character is present in the first line of each segment. Otherwise, segments are merged with their preceding segment. The concept key is extracted from text is the portion before the colon character in the first line of each segment, or from text in capital letters at the beginning of each segment. In cases where no concept key is found, we assign the concept as $Uncategorized_i$ where i enumerates uncategorized cases. The first segment's concept is set as $Start$ if no concept is identified using the aforementioned method.

A similar methodology is applied to the "brief_hospital_course" text, where we lack a predefined list. After denoising the text, we split the document as before and retain segments if the first line contains a colon, starts with a hash or greater than sign (">"), or is in all capital letters. The concept for each segment is determined by text preceding a colon, dash, or full stop sign in the first line, or by text in all capital letters.

3.3 Summarization

The primary objective of our summarization process is to condense document size while retaining crucial information. To achieve this goal, we avoid running the summarization model on the entire document due to the risk of potential loss of important details, especially in longer documents. Instead, we employ a specific prompt before each segment, which is given in Table 3 of Appx. A.3.

The summarization model is then applied solely to the source text and to each text segment from both the discharge and radiology reports. The summaries generated for each segment corresponding to each report are combined, resulting in a new concise report for each type of report. This helps reduce the document size with minimal information loss.

3.4 Concept Generation

Concepts play a pivotal role in our generation task, as they define the structure of the generated text for each query. To facilitate this, we train a Large Language Model (LLM) for each task—generating "brief_hospital_course" and "discharge_instructions". These models take the condensed versions of discharge and radiology texts

Metric	Performance
BLEU	0.04
ROUGE-1	0.21
ROUGE-2	0.1
ROUGE-L	0.13
BERTScore	0.18
Meteor	0.30
AlignScore	0.20
MEDCON	0.19
Overall	0.17

Table 1: Performance of Proposed model on Phase-2 test data

as input and generate a list of concepts extracted through our earlier methods.

In our approach, the prompt (Table 4 of Appx. A.4) provided to the model includes the summarized text along with a query aimed at identifying all concepts present in the text. The model's response provides all identified concepts, with each concept listed on a new line for clarity and organization. This method ensures that the generated text aligns with the extracted concepts, shaping the output according to the underlying structure of the input data.

3.5 Concept Based Description Generation

The concepts we extract from the previous section serve as directives for our model, guiding it to generate concise and relevant descriptions. These extracted concepts provide a roadmap for the generator, indicating the specific topic it should focus on. Our approach involves training a model that takes summarized text as input, along with a comprehensive list of concepts extracted using the method described earlier. We then train a Large Language Model (LLM) for each task, with each task designed to answer a question based on the input text.

We provide response as in Table 4 of Appx. A.4. The model's response is expected to be a description of that particular concept only. Once we have obtained all concepts and their corresponding descriptions for each task, we combine them to generate the final output text. In the combination process, we can exclude concepts such as "uncategorized" or "start" to enhance the naturalness of the output. This method allows us to generate well-structured and explanatory output for each task, resulting in a coherent and informative final text.

4 Results and Discussion

The "Discharge Me!" dataset comprises training (68,785 samples), validation (14,719 samples), phase I testing (14,702 samples), and phase II testing (10,962 samples) datasets, all sourced from MIMIC-IV submodules. It is worth noting that the phase II testing dataset, set for release on April 12th, 2024, will serve as the final evaluation test set. Due to time constraints, we were unable to execute our model on the entire dataset, ultimately utilizing a subset of 10,000 samples for training purposes. During the generation of summaries using the BioMistral-7B model (Labrak et al., 2024), we set the temperature to 0.0 to ensure determinism in our results. This controlled setting aimed to maintain consistency in the generated outputs during our experiments.

We proceeded to fine-tune the same model separately for four distinct tasks: concept generation and subsequent description generation corresponding to each concept, aimed at generating the "Brief Hospital Course" and "Discharge Instructions" sections. This fine-tuning process involved adopting the instruction tuning method, implemented using the Supervised Fine-tuning Trainer [1]. We configured the parameters as follows: $max\_seq\_length = 4096$, $learning\_rate = 2e-4$. To reduce memory size, we utilized 4-bit quantization, and for Low-Rank Adaptation (Mangrulkar et al., 2022), we set $rank = 64$, $alpha = 16$, and $dropout = 0.1$. The model underwent training for 5 epochs, employing a training batch size of 4 and an evaluation batch size of 20. These parameter settings were chosen to strike a balance between training efficiency and model performance across the different tasks.

In Table 1, we present the metrics proposed by the task organizers, which were calculated based on the outputs generated by our model. These metrics encompass a comprehensive evaluation of the model's performance across various dimensions specified by the Discharge Me task. The calculations were performed using Codabench. Additionally, for transparency and reproducibility, the task organizers have provided a Python script[2] for scoring, ensuring consistency and facilitating further analysis of our model's results.

[1] https://huggingface.co/docs/trl/en/sft_trainer

[2] https://github.com/Stanford-AIMI/discharge-me/tree/main/scoring

5 Conclusion

In conclusion, this paper describes our participation in the DischargeMe shared task, which involved generating summaries of hospital course and discharge reports using MIMIC-IV data. Our approach included data segmentation, concept identification and description, prompt-based summarization, and training models for concept extraction and description generation. We used pre-trained and fine-tuned Large Language Models (LLMs) to produce structured, informative summaries. Future work will compare different models and prompts and explore advanced data segmentation techniques to improve accuracy and efficiency.

6 Limitations

Our work faces challenges such as the necessity to summarize each text segment, limitations of rule-based methods, handling long segments with threshold limits, and dependence on the model and prompt used. Time constraints have hindered comprehensive comparisons of different models and prompts. We plan to develop a new model for better segmenting lengthy documents. The success of our generation task depends on accurate concept generation, as poor summarization impacts overall quality. These challenges highlight the complexity of the task and need for ongoing research and improvement.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Tal Baumel, Matan Eyal, and Michael Elhadad. 2018. Query Focused Abstractive Summarization: Incorporating Query Relevance, Multi-Document Coverage, and Summary Length Constraints into seq2seq Models. *arXiv preprint*. ArXiv:1801.07704 [cs].

Hal Daumé III and Daniel Marcu. 2006. Bayesian Query-Focused Summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 305–312, Sydney, Australia. Association for Computational Linguistics.

Sunyang Fu, David Chen, Huan He, Sijia Liu, Sungrim Moon, Kevin J. Peterson, Feichen Shen, Liwei Wang, Yanshan Wang, Andrew Wen, Yiqing Zhao, Sunghwan Sohn, and Hongfang Liu. 2020. Clinical Concept Extraction: A Methodology Review. *Journal of Biomedical Informatics*, 109:103526.

Quentin Grail, Julien Perez, and Eric Gaussier. 2021. Globalizing BERT-based Transformer Architectures for Long Document Summarization. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1792–1810, Online. Association for Computational Linguistics.

Luyang Huang, Shuyang Cao, Nikolaus Parulian, Heng Ji, and Lu Wang. 2021. Efficient Attentions for Long Document Summarization. *arXiv preprint*. ArXiv:2104.02112 [cs].

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Leo Anthony Celi, Roger Mark, and Steven Horng. a. MIMIC-IV-ED.

Alistair Johnson, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. b. MIMIC-IV-Note: Deidentified Free-text Clinical Notes.

Alistair Johnson, Tom Pollard, and Roger Mark. 2015. MIMIC-III Clinical Database.

Huan Yee Koh, Jiaxin Ju, Ming Liu, and Shirui Pan. 2022. An Empirical Survey on Long Document Summarization: Datasets, Models, and Metrics. *ACM Computing Surveys*, 55(8):154:1–154:35.

Sunjun Kweon, Junu Kim, Jiyoun Kim, Sujeong Im, Eunbyeol Cho, Seongsu Bae, Jungwoo Oh, Gyubok Lee, Jong Hak Moon, Seng Chan You, Seungjin Baek, Chang Hoon Han, Yoon Bin Jung, Yohan Jo, and Edward Choi. 2024. Asclepius-R : Clinical Large Language Model Built On MIMIC-III Discharge Summaries.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. Biomistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains. *Preprint*, arXiv:2402.10373.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. Peft: State-of-the-art Parameter-Efficient Fine-Tuning Methods. `https://github.com/huggingface/peft`.

Gianluca Moro and Luca Ragazzi. 2022. Semantic Self-Segmentation for Abstractive Summarization of Long Documents in Low-resource Regimes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11085–11093.

Bo Pang, Erik Nijkamp, Wojciech Kryscinski, Silvio Savarese, Yingbo Zhou, and Caiming Xiong. 2023. Long Document Summarization with Top-down and Bottom-up Inference. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1267–1284, Dubrovnik, Croatia. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Ramakrishna Varadarajan and Vagelis Hristidis. 2006. A System for Query-specific Document Summarization. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, CIKM '06, pages 622–631, New York, NY, USA. Association for Computing Machinery.

Jesse Vig, Alexander Fabbri, Wojciech Kryscinski, Chien-Sheng Wu, and Wenhao Liu. 2022. Exploring Neural Models for Query-Focused Summarization. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1455–1468, Seattle, United States. Association for Computational Linguistics.

Justin Xu, Zhihong Chen, Andrew Johnston, Louis Blankemeier, Maya Varma, Jason Hom, William J. Collins, Ankit Modi, Robert Lloyd, Benjamin Hopkins, Curtis Langlotz, and Jean-Benoit Delbrouck. 2024. Overview of the First Shared Task on Clinical Text Generation: RRG24 and "Discharge Me!" In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a Novel Ambient Clinical Intelligence Dataset for Benchmarking Automatic Visit Note Generation. *Scientific Data*, 10(1):586. Publisher: Nature Publishing Group.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating Factual Consistency with A Unified Alignment Function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. *arXiv preprint*. ArXiv:1904.09675 [cs].

A Appendix

A.1 Source and Target Text Segments

As described in Sections 3.1 and 3.2, Figures 2 and 3 illustrate the segmentation of source and target texts, respectively. In Figure 2, different segments are denoted by various colors. In Figure 3, all segments are separated, with the blue parts representing the concepts and the corresponding text in yellow serving as their descriptions.

A.2 Training Data Statistics

The training data statistics are presented in Table 2. It shows the minimum, maximum, and average document token lengths for both source and target texts arranged against "hadm_id". The table also categorizes the documents based on token length into three groups: less than 500, between 500 and 2000, and more than 2000 tokens. From the table, we can observe that the source discharge texts are typically very long, whereas the target discharge instruction texts are usually short.

A.3 Prompt for Summarization

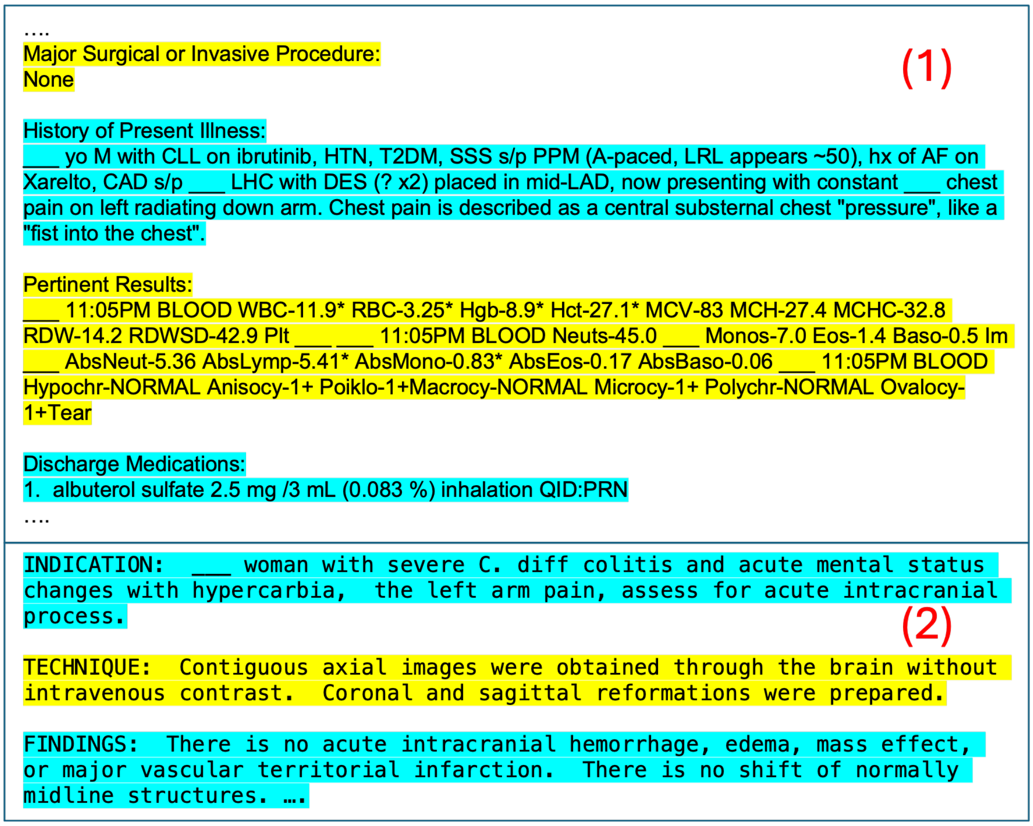
In Table 3, we provide the prompt used for summarizing all text segments to ensure minimal loss of important information. We use this prompt with BioMistral-7B-DARE to generate summaries for each segment.

A.4 Prompt for Fine-tuning Model

In Table 4, we provide the prompt used to train the LLM to generate the "Concept" and the corresponding description. This table depicts two prompt templates. The first template is used for extracting concepts from the summarized source text. The second template generates a description of a particular concept given the same summarized source text and the list of all extracted concepts.

	Min token length	Max token length	Avg token length	<500	>500 <2000	>2000
Radiology	34	48980	1491.7	19762	34184	14839
Discharge	510	21087	4263	0	3186	65599
Discharge instruction	12	8935	332.6	57328	11378	79
Brief hospital course	13	6959	635.6	32068	35654	1063

Table 2: Data statistics for the given training corpus, calculated using Mistral Tokenizer (Labrak et al., 2024)



....
Major Surgical or Invasive Procedure:
None

History of Present Illness:
___ yo M with CLL on ibrutinib, HTN, T2DM, SSS s/p PPM (A-paced, LRL appears ~50), hx of AF on Xarelto, CAD s/p ___ LHC with DES (? x2) placed in mid-LAD, now presenting with constant ___ chest pain on left radiating down arm. Chest pain is described as a central substernal chest "pressure", like a "fist into the chest".

Pertinent Results:
___ 11:05PM BLOOD WBC-11.9* RBC-3.25* Hgb-8.9* Hct-27.1* MCV-83 MCH-27.4 MCHC-32.8 RDW-14.2 RDWSD-42.9 Plt ___ ___ 11:05PM BLOOD Neuts-45.0 ___ Monos-7.0 Eos-1.4 Baso-0.5 Im ___ AbsNeut-5.36 AbsLymp-5.41* AbsMono-0.83* AbsEos-0.17 AbsBaso-0.06 ___ 11:05PM BLOOD Hypochr-NORMAL Anisocy-1+ Poiklo-1+Macrocy-NORMAL Microcy-1+ Polychr-NORMAL Ovalocy-1+Tear

Discharge Medications:
1. albuterol sulfate 2.5 mg /3 mL (0.083 %) inhalation QID:PRN
....

INDICATION: ___ woman with severe C. diff colitis and acute mental status changes with hypercarbia, the left arm pain, assess for acute intracranial process.

TECHNIQUE: Contiguous axial images were obtained through the brain without intravenous contrast. Coronal and sagittal reformations were prepared.

FINDINGS: There is no acute intracranial hemorrhage, edema, mass effect, or major vascular territorial infarction. There is no shift of normally midline structures. ….

Figure 2: The rule based text segmentation for 1. *Discharge text* 2. *Radiology text*. Two consecutive segments are marked by different colors.

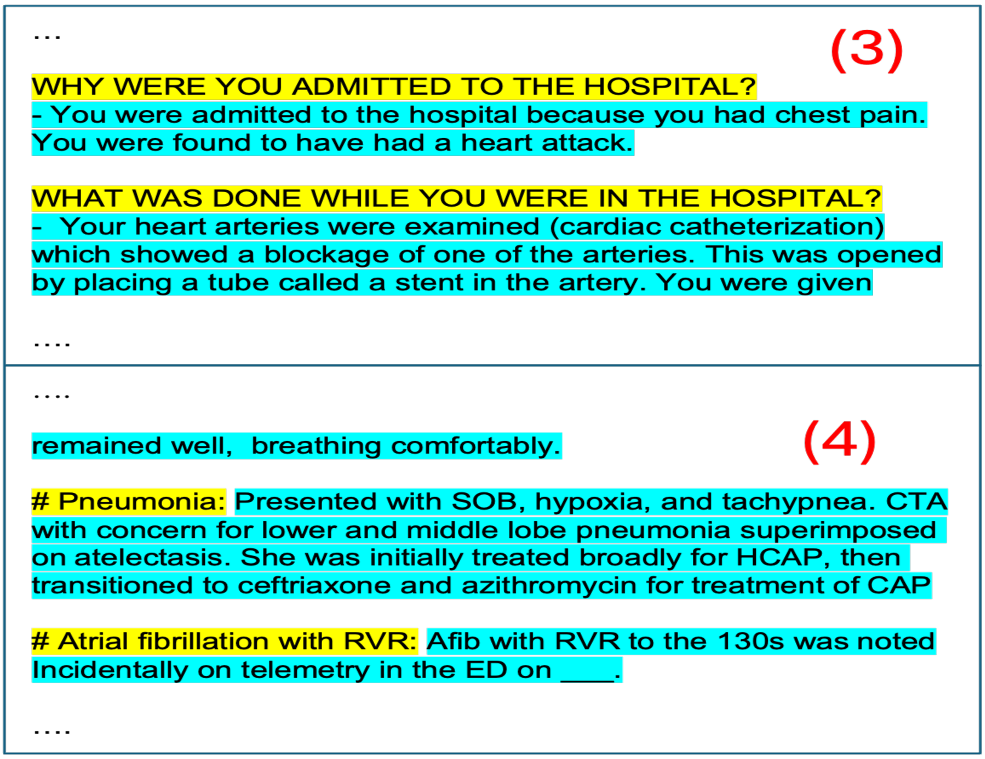
Prompt for Summarization

You are an intelligent clinical language model.
Below is a snippet of patient's discharge summary and a following instruction from healthcare professional.
Write a response that appropriately completes the instruction.
The response should provide the accurate answer to the instruction, while being concise.

[Discharge Report Begin]
{text_segment}
[Discharge Report End]

[Instruction Begin]
Summarize the text in very concise form, only keep the important information.
[Instruction End]

Table 3: Prompt for summarization task of each text segment



...

WHY WERE YOU ADMITTED TO THE HOSPITAL?
- You were admitted to the hospital because you had chest pain. You were found to have had a heart attack.

WHAT WAS DONE WHILE YOU WERE IN THE HOSPITAL?
- Your heart arteries were examined (cardiac catheterization) which showed a blockage of one of the arteries. This was opened by placing a tube called a stent in the artery. You were given

....

....

remained well, breathing comfortably.

Pneumonia: Presented with SOB, hypoxia, and tachypnea. CTA with concern for lower and middle lobe pneumonia superimposed on atelectasis. She was initially treated broadly for HCAP, then transitioned to ceftriaxone and azithromycin for treatment of CAP

Atrial fibrillation with RVR: Afib with RVR to the 130s was noted Incidentally on telemetry in the ED on ___.

....

Figure 3: The rule based text segmentation for 3. *Discharge Instruction* 4. *Brief Hospital Course*. The Concepts are marked in yellow followed by corresponding description in blue.

Prompt for Extraction of Concept
Instruction: Below is a input context which contains the summaries of discharge and radiology reports followed by a question. Generate the response for the question using the context. ### Input: {Summarised Source Text} ### Question: What are the possible aspects for {discharge instruction / brief hospital course} in the above document?
Prompt for Generation of Description Corresponding to Concept
Instruction: Below is a input context which contains the summaries of discharge and radiology reports followed by a question. Generate the response for the question using the context. ### Input:{Summarised Source Text} ### Concepts: {List of Concepts} ### Question: Describe the concept C_i based on the above text.

Table 4: Prompt Template for Fine-tune a LLM to generate Concept and Corresponding Description

DeakinNLP at BioLaySumm: Evaluating Fine-tuning Longformer and GPT-4 Prompting for Biomedical Lay Summarization

Huy Quoc To, Ming Liu, Guangyan Huang
School of Information Technolgy, Deakin University, Australia
{q.to, m.liu, guangyan.huang}@deakin.edu.au

Abstract

This paper presents our approaches for the BioLaySumm 2024 Shared Task. We evaluate two methods for generating lay summaries based on biomedical articles: (1) fine-tuning the Longformer-Encoder-Decoder (LED) model, and (2) zero-shot and few-shot prompting on GPT-4. In the fine-tuning approach, we individually fine-tune the LED model using two datasets: PLOS and eLife. This process is conducted under two different settings: one utilizing 50% of the training dataset, and the other utilizing the entire 100% of the training dataset. We compare the results of both methods with GPT-4 in zero-shot and few-shot prompting. The experiment results demonstrate that fine-tuning with 100% of the training data achieves better performance than prompting with GPT-4. However, under data scarcity circumstances, prompting GPT-4 seems to be a better solution.

1 Introduction

The task of summarization has witnessed the development based on pre-trained language models. More recently, the superiority of large language models (LLMs) has been demonstrated on a wide range of natural language processing (NLP) tasks (Minaee et al., 2024; Zhao et al., 2023). In the BioLaySumm 2024 shared task (Goldsack et al., 2024), the competition focuses on generating summaries for biomedical research articles that are easily understandable by the general public. These summaries are usually known as "lay summaries".

Recently, the study of the summarization task using generative models has increased for both general domains (Koh et al., 2022b; Zhao et al., 2020) and biomedical text (Liu et al., 2023a). Additionally, according to Goldsack et al. (2022), each article generally has more than 10,000 words. Many pre-trained language models have been developed to handle such long text (Koh et al., 2022a). In this paper, we implement the Longformer-Encoder-Decoder (LED) (Beltagy et al., 2020) as an approach for Biolaysumm shared task, as its performance has been demonstrated in (Liu et al., 2023b; Wu et al., 2023).

In this paper, we present a comparison between the performance of the fine-tuned LED model on 50% and 100% of the training set. Additionally, we evaluate GPT-4 (OpenAI et al., 2024) on zero-shot and few-shot prompting for this Shared Task. Our aim is to investigate how a fine-tuned model and a large language model such as GPT-4 perform in lay summarization biomedical text. This study focuses on three aspects: performance, training time, and computational cost. Our contributions are as follows.

- We fine-tune LED model on different amount of data to evaluate how it affects the performance of the LED model in biomedical lay summarization task.

- Secondly, we evaluate GPT-4 on zero-shot and few-shot prompting to investigate how the in-context learning capability of this model. Our results show that, in the eLife dataset, the GPT-4 few-shot prompting method outperforms the fine-tuned LED model.

In the following sections, we briefly analyze the datasets, describe our methods in detail, showcase the experiment settings, and present our results, findings, and conclusion.

2 Datasets

The task is evaluated on two datasets: PLOS and eLife (Goldsack et al., 2022). Both datasets contain biomedical articles and a lay summary manually written for each article. To first understand the evaluation datasets, we proceed tokenizing the input and the output text on two datasets using tokenizer from LED model (Beltagy et al., 2020). We sum-

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 748–754
August 16, 2024. ©2024 Association for Computational Linguistics

marize the statistics of the PLOS and eLife dataset in Table 1.

Dataset	**Article**(#Tokens)			**Summ.**(#Tokens)	
	Train	Val	Test	Train	Val
PLOS	9,851	9,924	9,978	263	279
eLife	12,321	12,753	11,967	435	445

Table 1: The mean number of tokens of input and output text in PLOS and eLife datasets. **Summ.** is the abbreviation for lay summary.

According to (Goldsack et al., 2024) and Table 1, while PLOS has more instances of biomedical papers than the eLife dataset, and the length of both input and output text in eLife is longer than PLOS. We also notice that the maximum number of tokens for input text is 28,561 for PLOS and 34,612 tokens in eLife.

3 Evaluation Metrics

In this shared task, the generated summaries are evaluated on three aspects and ten metrics accordingly:

- **Relevance**: ROUGE-1 (R-1), ROUGE-2 (R-2) and ROUGE-L (R-L) (Lin, 2004) and BERTScore (Zhang et al., 2020).
- **Readability**: Flesch-Kincaid Grade Level (FKGL) (Kincaid et al., 1975) and Dale-Chall Readability Score (DCRS) (Chall and Dale, 1995), Coleman-Liau Index (CLI), and LENS (Maddela et al., 2023).
- **Factuality** - AlignScore (Zha et al., 2023), and SummaC (Laban et al., 2022).

The objective of the evaluation is to maximize the Relevance, Factuality and LENS in Readability scores and minimize FKGL, DCRS, and CLI scores.

4 Preliminary Study

Datasets	**Section**	**R-1**	**R-2**	**R-L**	**BertScore**
PLOS	Abs.	**0.502**	**0.199**	**0.466**	**0.871**
	Con.	0.154	0.039	0.146	0.803
	Others	0.084	0.041	0.081	0.832
eLife	Abs.	**0.319**	**0.071**	**0.293**	**0.839**
	Con.	0.162	0.026	0.156	0.782
	Others	0.097	0.033	0.095	0.820

Table 2: Analysis on Relevance aspect of the abstract, conclusion and the rest of the content with the lay summaries.

Due to the fact that each instance in both datasets is lengthy and may contain a large amount of irrelevant information to generate lay summaries, we perform a heuristic evaluation on the validation sets. We are aware that each article has at least an abstract and a conclusion paragraph. We evaluated the abstract, the conclusion part and other parts of each article with the lay summaries on the **Relevance** aspect. Table 2 shows that in both cases, the abstracts written by the author of each article contain the most similar information. These abstracts are likely to be used as the base knowledge when creating the lay summaries. Additionally, the conclusion parts also achieve competitive scores, which indicates that they have potential to be used as sources to generate lay summaries.

5 Experiments

Based on the results of our preliminary study, we first extract the abstract and conclusion paragraph from the original articles. We then perform the fine-tuning process and prompting GPT-4 using the combination of abstract and conclusion from the original articles.

5.1 Fine-tuning LED model

We fine-tune LED model on each dataset individually using 50% and 100% of the training set. We randomly select 50% of the training instances. Fine-tuning processes are performed on Colab Pro [1] using the L4 GPU (22GB VRam). We employ the base version (41M parameters) of the LED model via Huggingface[2], which can process up to 16,384 tokens. In the experiment, the batch size is set to 2 due to the limitation of the GPU VRam, and we train for 2 epochs and set the learning rate to 1e-5. For the PLOS dataset, we set the maximum token at 10,000 for input, and the maximum output sequence length is 400 tokens. Since the eLife dataset has longer input and output sequence lengths, we set the maximum input token to 14,000 tokens, and the output is 600 tokens. These adjustments are made to accommodate the length of the lay summary in each dataset.

[1] https://colab.research.google.com/

[2] https://huggingface.co/docs/transformers/en/model_doc/longformer

Model	**R-1**	**R-2**	**R-L**	**BertScore**	**FKGL**	**DCRS**	**CLI**	**LENS**	**AlignScore**	**SummaC**
					PLOS					
LED (50%)	0.472	0.157	0.426	0.864	14.459	11.431	15.781	56.053	0.818	**0.741**
LED (100%)	**0.472**	**0.163**	**0.431**	**0.865**	14.299	11.367	15.520	57.090	**0.819**	0.739
GPT-4 zs	0.420	0.114	0.385	0.857	14.648	10.556	15.456	70.621	0.646	0.485
GPT-4 fs	0.431	0.123	0.402	0.860	**14.210**	**10.530**	**15.380**	**70.781**	0.711	0.589
					eLife					
LED (50%)	0.456	0.121	0.435	0.843	9.456	7.760	10.351	67.392	0.631	0.601
LED (100%)	0.461	0.121	0.441	0.848	**9.448**	**7.752**	**10.345**	68.453	0.653	**0.617**
GPT-4 zs	0.465	0.101	0.431	0.847	15.320	10.707	16.641	68.769	0.656	0.477
GPT-4 fs	**0.493**	**0.121**	**0.457**	**0.851**	14.626	10.145	15.435	**70.732**	**0.672**	0.497

Table 3: The performance of the evaluated models on the PLOS and eLife private test sets. The best score for each metric is highlighted in bold, and the second-best score is underlined. ZS is short for zero-shot and FS is short for few-shot.

Model	**R-1**	**R-2**	**R-L**	**BertScore**	**FKLG**	**DCRS**	**CLI**	**LENS**	**AlignScore**	**SummaC**
BART (Baseline)	0.470	0.140	0.436	**0.862**	**12.035**	**10.147**	**13.485**	48.096	**0.779**	**0.703**
Final Submission	**0.482**	**0.142**	**0.444**	0.858	14.462	10.755	15.477	**63.912**	0.745	0.618

Table 4: Our final submission is the combination of fine-tuned 100% training set LED model on PLOS dataset and GPT-4 few-shot prompting on eLife dataset.

5.2 Prompting GPT-4

GPT-4 demonstrates strong performance on few-shot settings in multiple NLP tasks (Liu et al., 2023c). In our experiments, we access GPT-4 through OpenAI APIs[3]. To save cost, we choose **gpt-4-turbo-preview** version to generate lay summaries. We evaluated GPT-4 in two settings: zero-shot and few-shot prompting. In zero-shot prompting, we directly pass the extracted input to GPT-4 and generate the lay summaries. When creating prompts in few-shot settings, we randomly pick the source-target pairs from the validation set and use them as examples for GPT-4. Since the maximum tokens that GPT-4 can take are 128,000 token, we incorporate as many as possible within the token constraints of the API calls. As the results, PLOS and eLife few-shot prompts contain 4 and 3 example pairs, respectively. The maximum lay summary length is set to 400 tokens and 600 tokens, respectively, for PLOS and eLife. We present an example of a zero-shot prompt and a few-shot prompt in Appendix A.

6 Results

In this section, we list our results on the private test set. The scores are retrieved through the Codabench page of the shared task and reported in Table 3.

PLOS The results clearly demonstrate that fine-tuning the LED model achieves the best performance on relevance and factual aspects. To our surprise, GPT-4 outperforms LED in readability. The FKGL score of the fine-tuned LED model with 100% train set achieves the second best results. However, for other readability metrics, the performance of LED models is worse than GPT-4 prompting. In particular, the gap in the LENS score is noticeably high. The gap is around 13.6 percentage points when comparing the fine-tuned version of LED (100%) with GPT-4 few-shot prompting. Meanwhile, compared to the results of the GPT-4 few-shot prompting, the fine-tuned LED model with full training data outperforms by 0.041, 0.039, 0.029, 0.005, 0.108, and 0.150 on R-1, R-2, R-L, BERTScore, AlignScore, and SummaC, respectively. It seems that the improvement of the best fine-tuned LED on those scores can be considered marginal.

eLife On the eLife dataset, it is surprising that GPT-4 outperforms fine-tuned LED model in generating more accurate summaries. However, the difference in readability is significant, as GPT-4 achieves lower scores on FKGL, DCRS, and CLI compared to LED models. The gaps between GPT-4 and LED model on these three metrics, respec-

[3] https://platform.openai.com/docs/models/gpt-4-turbo-and-gpt-4

tively, are 5.178, 2.393, 5.090. Whereas, the differences that GPT-4 few-shot prompting creates compared to LED (100%) fine-tuned version on R-1, R-2, R-L, BertScore, LENS, and AlignScore, respectively, are 0.032, 0, 0.016, 0.003, 2.279, and 0.019. It is no doubt that on eLife dataset, prompting GPT-4 generates better lay summaries in terms of Relevance and Factuality.

Based on the above results, we made our final submission to the shared task by combining the results of the fine-tuned LED model with 100% training data from PLOS and GPT-4 few-shot prompts in the eLife dataset. We compare our submission with the BART baseline (Goldsack et al., 2024) in Table 4. It shows that our results surpass the baseline on the R-1, R-2, R-L, and LENS scores. Remarkably, our LENS score is higher than BART baseline by 15.816%. Although in the other metrics, our results are a bit lower than baseline, we argue that the scores are still competitive and the gap is marginal.

7 Discussion

The results demonstrate that traditional fine-tuning can produce summaries with accurate keywords and context rather than prompting. LED model also creates less hallucination than LLMs, because it achieves better Factuality scores. However, fine-tuning is less effective in making the summaries simpler and easier to understand.

Furthermore, we believe that fine-tuning LED model on eLife is less efficient than on PLOS dataset because of the size of eLife dataset. Furthermore, the text in eLife dataset is also longer than PLOS. Therefore, it is likely that LED model is not able to capture the keywords and learn enough context on eLife. Hence, GPT-4's performance is slightly better in this case.

8 Performance Versus Cost

In this section, we discuss the trade-off between model performance and costs. In our analysis, the costs include training time, computational cost, and prompting cost. We summarize our comparison in Table 5. We first rank the performance of each method based on the results in Table 3. Next, we evaluate four methods based on the number of training hours, the costs of training, inference, and prompting. Since the PLOS dataset has more instances in the training set than eLife, it undoubtedly takes more time and more costly to train LED models on PLOS. In Colab Pro[4], it costs around 5 computational units per hour. Hence, to calculate the total computational cost, we simply multiply 5 by the training time.

Model	#Rank	Training	Cost
	PLOS		
LED (50%)	*2nd*	8 hrs	40 units
LED (100%)	*1st*	20 hrs	100 units
GPT-4 zs	*4th*	0 hr	10$
GPT-4 fs	*3rd*	0 hr	20$
	eLife		
LED (50%)	*4th*	4 hrs	20 units
LED (100%)	*2nd*	8.5 hrs	42.5 units
GPT-4 zs	*3rd*	0 hr	20$
GPT-4 fs	*1st*	0 hr	30$

Table 5: The comparision between four approaches on two datasets. The cost for fine-tuning is referred to computation units and cost for GPT-4 is referred to prompting cost using OpenAI APIs.

On the other hand, we directly prompt GPT-4 without further fine-tuning the model. Therefore, we only report the prompting cost in two data sets. As mentioned in Table 1, the length of each instance in the eLife test set is longer than PLOS, and it costs more to generate the lay summaries. In the few-shot prompting setting, it also costs more because we include more tokens in the queries for example.

Through our result analysis and cost-effective study, it demonstrates that GPT-4 prompting cost us more on querying, however it takes less time then fine-tuning and still achieves competitive results. Especially, in the situation where we have less training data (such as in eLife case), GPT-4 can outperform fine-tuned LED model.

9 Conclusion

This paper details our approach to the BioLaySumm 2024 shared task, comparing traditional fine-tuning of the Longformer-Encoder-Decoder (LED) model and few-shot prompting with GPT-4 for generating lay summaries of biomedical articles. Our results indicate that the fine-tuned LED excels on the PLOS dataset, while GPT-4's few-shot prompting outperforms LED on the eLife dataset, highlighting GPT-4's advantage in data scarcity scenarios. Future work may explore self-evaluation meth-

[4] In 2024, 100 computational units cost around 15$ on Colab Pro.

ods and cost-reduction strategies for fine-tuning using parameter-efficient techniques.

10 Limitations

Our methodology relies exclusively on OpenAI APIs for generating summaries using GPT-4, which presents minimal technical challenges. However, the costs associated with API requests can quickly escalate to prohibitive levels, limiting our ability to conduct extensive experimental work with the model. Implementing proprietary LLMs such as GPT-4 also has the limitations of reproducing the results. In addition, due to computational cost and time constraints, we were unable to fine-tune the LED model for more epochs, potentially impacting the overall performance.

References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *Preprint*, arXiv:2004.05150.

J.S. Chall and E. Dale. 1995. *Readability Revisited: The New Dale-Chall Readability Formula*. Brookline Books.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Peter Kincaid, Robert P. Fishburne, Richard L. Rogers, and Brad S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Huan Yee Koh, Jiaxin Ju, Ming Liu, and Shirui Pan. 2022a. An empirical survey on long document summarization: Datasets, models, and metrics. *ACM Comput. Surv.*, 55(8).

Huan Yee Koh, Jiaxin Ju, He Zhang, Ming Liu, and Shirui Pan. 2022b. How far are we from robust long abstractive summarization? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2682–2698, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Ming Liu, Dan Zhang, Weicong Tan, and He Zhang. 2023a. DeakinNLP at ProbSum 2023: Clinical progress note summarization with rules and language ModelsClinical progress note summarization with rules and languague models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 491–496, Toronto, Canada. Association for Computational Linguistics.

Quancheng Liu, Xiheng Ren, and V.G.Vinod Vydiswaran. 2023b. LHS712EE at BioLaySumm 2023: Using BART and LED to summarize biomedical research articles. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 620–624, Toronto, Canada. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023c. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. Lens: A learnable evaluation metric for text simplification. *Preprint*, arXiv:2212.09739.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large language models: A survey. *Preprint*, arXiv:2402.06196.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Yu-Hsuan Wu, Ying-Jia Lin, and Hung-Yu Kao. 2023. IKM_Lab at BioLaySumm task 1: Longformer-based prompt tuning for biomedical lay summary generation. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 602–610, Toronto, Canada. Association for Computational Linguistics.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

Jinming Zhao, Ming Liu, Longxiang Gao, Yuan Jin, Lan Du, He Zhao, He Zhang, and Gholamreza Haffari. 2020. Summpip: Unsupervised multi-document summarization with sentence graph compression. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 1949–1952, New York, NY, USA. Association for Computing Machinery.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models. *Preprint*, arXiv:2303.18223.

A Example prompts on GPT-4

	Zero-shot prompt
role	*system*
content	"Write a lay summary using the following research abstract and conclusion."
role	*user*
content	"Lung-resident (LR) mesenchymal stem and stromal cells (MSCs) are key elements of the alveolar niche and fundamental regulators of homeostasis and regeneration..."
	Few-shot prompt
role	*system*
content	"Write a lay summary using the following research abstract and conclusion."
role	*user*
content	"Gene expression varies widely between individuals of a population , and regulatory change can underlie phenotypes of evolutionary and biomedical relevance..."
role	*assistant*
content	"Messenger RNAs carry the instructions necessary to synthesize proteins that do work for the cell..."
role	*user*
content	"The live attenuated simian immunodeficiency virus (LASIV) vaccine SIVnef is one of the most effective vaccines..."
role	*assisstant*
content	"Annually, more than two million people are infected with HIV , the virus that causes AIDS..."
role	*user*
content	"Mucosal infections with Candida albicans belong to the most frequent forms of fungal diseases..."
role	*assisstant*
content	"The opportunistic pathogen Candida albicans is a major risk factor for immunosuppressed individuals..."
role	*user*
content	"Lung-resident (LR) mesenchymal stem and stromal cells (MSCs) are key elements of the alveolar niche and fundamental regulators of homeostasis and regeneration..."

Table 6: Example of zero-shot prompt and few-shot prompt for GPT-4.

ELiRF-VRAIN at BioLaySumm: Boosting Lay Summarization Systems Performance with Ranking Models

Vicent Ahuir[†], **Diego Torres**[†,*], **Encarna Segarra**[†,§], **Lluís-F. Hurtado**[†]
[†]VRAIN: Valencian Research Institute for Artificial Intelligence
Universitat Politècnica de València, Spain
[§]ValgrAI: Valencian Graduate School and Research Network of Artificial Intelligence
[†]{vahuir,esegarra,lhurtado}@dsic.upv.es
[*]dtorber@etsinf.upv.es

Abstract

This paper presents our contribution to the BioLaySumm 2024 shared task of the 23rd BioNLP Workshop. The task is to create a lay summary, given a biomedical research article and its technical summary. As the input to the system could be large, a Longformer Encoder-Decoder (LED) has been used. We continuously pre-trained a general domain LED model with biomedical data to adapt it to this specific domain. In the pre-training phase, several pre-training tasks were aggregated to inject linguistic knowledge and increase the abstractivity of the generated summaries. Since the distribution of samples between the two datasets, eLife and PLOS, is unbalanced, we fine-tuned two models: one for eLife and another for PLOS. To increase the quality of the lay summaries of the system, we developed a regression model that helps us rank the summaries generated by the summarization models. This regression model predicts the quality of the summary in three different aspects: *Relevance*, *Readability*, and *Factuality*. We present the results of our models and a study to measure the ranking capabilities of the regression model.

1 Introduction

Nowadays, there is more information than ever at the disposal of the general public. In the specific domain of biomedical research, there is information that would be interesting to non-expert audiences, including journalists or even members of the public, such as what occurred during the recent COVID-19 global pandemic (Wang et al., 2020). However, the technical language is a barrier for the non-specialist public that may prevent them from accessing that information (Goldsack et al., 2022; Guo et al., 2021).

Abstract summarization models should be useful in reducing the gap in understanding information. Since the models can generate a concise summary of a given text and capture its most relevant information (Raffel et al., 2020; Lewis et al., 2020; Brown et al., 2020; Beltagy et al., 2020). It is possible to obtain new models that generate summaries adapted to a much wider audience; what is known as *lay summary*. In a lay summary, the text should contain the main ideas of the article that would be interesting for a non-expert audience, enhancing readability by adding background information and reducing (or avoiding) technical terminology.

In this paper, we present the results and analysis of our system in the participation at the BioLaySumm (Goldsack et al., 2024) at the 23rd BioNLP Workshop (Demner-Fushman et al., 2024).

2 Task Description

In the 2024 edition, the BioLaySumm poses a single shared task, rather than two, as in the previous edition (Goldsack et al., 2023). The task is to create a lay summary, given a biomedical research article and its technical summary (abstract section of the article).

The organization provides a biomedical dataset (Goldsack et al., 2022) that contains biomedical research articles from two sources: *eLife Sciences*[1] and *Public Library of Science* (PLOS)[2]. Each sample contains the text of the article, the technical summary, and the reference lay summary. The dataset is divided into three partitions: `train`, `val`, and `test`.

	`train`	`val`	`test`
eLife	4346 (91.9)	241 (5.1)	142 (3.0)
PLOS	24 773 (94.3)	1376 (5.2)	142 (0.5)

Table 1: Dataset samples distribution per partition and source. Additionally to the number of samples, the table also shows the percentage over the source.

Table 1 shows the sample distribution of each

[1] https://elifesciences.org/
[2] https://plos.org

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 755–761
August 16, 2024. ©2024 Association for Computational Linguistics

source. It can be observed that the number of samples is way unbalanced towards the PLOS source, even though `test` presents the same number of samples for each source. This kind of distribution would be challenging when someone would like to develop a single summarization model without prompting or instructions. The alternative would be to create separate summarization models, one for eLife and the other for PLOS. The BioLaySumm organizers invited the participants to present solutions indistinctly using one or two models.

To measure the performance of the systems, the organizers of the competition selected a set of measures that would help to evaluate the performance in three different aspects: *Relevance*, *Readability*, and *Factuality*. For ***Relevance*** the following scores were chosen: ROUGE (1, 2, L) (Lin, 2004), BERTScore (Zhang* et al., 2020). To measure the ***Readability*** aspect: Flesch-Kincaid Grade Level (FKGL) (Kincaid et al., 1975), Dale-Chall Readability Score (DCRS) (Dale and Chall, 1948), Coleman-Liau Index (CLI) (Coleman and Liau, 1975), and LENS (Maddela et al., 2023). Finally, to measure ***Factuality***, two scores were selected: AlignScore (Zha et al., 2023), SummaC (Laban et al., 2022).

3 Pre-trainined Model

For this task, we have used a Longformer Encoder-Decoder (LED) (Beltagy et al., 2020) since we were approaching summarizing long texts, such as the case of scientific articles. This lets us increase the amount of information available on the encoder side. We used as a starting point the LED base model from AI2[3], publicly available at the repository of HuggingFace (Wolf et al., 2020), and continuously pre-trained it with in-domain data.

For the continual pre-training phase, we followed the training methodology used in the News Abstractive Summarization models (NAS) work (Ahuir et al., 2021). This methodology combines multiple pre-training tasks to incorporate linguistic knowledge in the pre-training phase and enhance the abstract nature of the produced summaries. Incorporating those tasks in continuous pre-training should help the model to transfer knowledge specific to the summarization task and increase the performance of the downstream model after fine-tuning, just as it did in the original NAS work.

The data used for continuous pre-training was chosen specifically to adapt the model to the biomedical research domain. We collected text from three different sources: abstracts (technical summaries) from PubMed (National Center for Biotechnology Information (NCBI), 2024) (17M samples), PubMed articles and abstracts from the `scientific_papers`[4] dataset (Cohan et al., 2018) (240K). Also, articles and technical summaries from the dataset train partition used in this competition (eLife + PLOS) (29K).

Due to infrastructure limitations, we limited the encoder input to work with no more than 4096 tokens. Taking into account this restriction, and with the objective of maximizing the amount of data, we split text by lines, using a window of no more than 4000 words. We generated subsamples that contained at least a new line and filled the windows with as many words as possible. The final amount of samples went up to 59M samples.

When working with LongFormers, you have to select which tokens will receive global attention in addition to local attention. In the original work (Beltagy et al., 2020), the authors recommend setting `[CLS]` token with global attention. However, we hypothesized that adding landmarks across the input with global attention could increase performance. For this reason, we added a special token with global attention (`<sent>`) after a certain number of sentences. The number of sentences was not constant but dictated by a minimum number of words of separation between `<sent>` tokens. Thus, the special token was placed at the end of every number of sentences with a total length of at least k words. Previous experimentation was carried out to determine the number of words. The best results were obtained with at least $k = 20$ words of separation.

The base model was pre-trained for three epochs in our Research Institute's cluster with 8 NVIDIA A40 graphic cards with 48GB of VRAM were used for the process; which took a month. The main hyperparameters are: 128 samples per device, 4 gradient accumulation steps, a learning rate of 5×10^{-5} with a constant scheduler, gradient checking, and an 8-bit quantified optimizer.

4 Lay Summarization Models

We developed two different approaches for the competition. In the first approach (**M1**), the model re-

[3] https://huggingface.co/allenai/led-base-16384

[4] http://tiny.cc/54x2yz/scientific_papers

ceives the technical summary and adapts the text and information to a lay summary style. In the second approach (**M2**), additional text is included beside the technical summary, that was, the introduction and the discussion sections of the article, similar to (Poornash et al., 2023).

Since the distribution of samples is not well-balanced, we fine-tuned two models per approach: one for eLife and another for PLOS. The four models were fine-tuned for ten epochs each with an NVIDIA RTX 3090 with 24GB; each approximation took nearly 24 hours. The relevant hyperparameters are: 4 samples per device and a learning rate of 5×10^{-5} with a linear scheduler.

In our tests over validation, M1 outperformed M2 in the overall performance. The detailed results can be seen in Table 3 (Appendix A).

5 Ranking Model

In order to increase the quality of the lay summaries of the system, we developed a regression model to rank the summaries generated by the summarization models. This regression model predicts the quality of the summary in three different aspects: *Relevance*, *Readability*, and *Factuality*.

5.1 Dataset Creation and Model Development

We use a Longformer encoder already trained in the biomedical domain[5] to develop the regression model. The classification layer was modified from the default in HuggingFace. We use a mean-max function of the hidden states of the last attention layer to calculate the embedding that feeds the feedforward classification layer. In mean-max, the mean of the hidden states is concatenated with the max values of those hidden states.

To fine-tune the model, we needed first to find a way to obtain sample variability in the scores in the three aspects. For this reason, we employed data augmentation based on LLMs. For this purpose, we adapted to our needs the novel framework *TextMachina* (Sarvazyan et al., 2024) and generated new samples using four LLMs: Vicuna 13b (Chiang et al., 2023), Alpaca 13b (Taori et al., 2023), OpenChat 7.5b (Wang et al., 2023), and Llama2 13b (Touvron et al., 2023). Using the technical summary and the lay summary from randomly selected samples of both sources, we applied different prompts to gain diversity in the quality of the samples in the three aspects. With this data augmentation, we obtained 16 236 new samples for training and 4212 for validation.

To create the training and validation partitions for regression, we use the generated samples and the technical and lay summaries from the corresponding partition of the competition dataset. To obtain the reference scores, we computed *Readability*, *Relevance*, and *Factuality*, using the formulas shown in Appendix B. At this point, we should remark on two details: (a) it can be noticed that all the scores are in a range $[0, 1]$, and always correlate positively with the quality of the summary, (b) due to time constraints, the *Factuality* score is only measured with *AlignScore* in the regression dataset.

The regression model was trained for five epochs in VRAIN's cluster for two days with 4 NVIDIA A30 with 24GB of VRAM. The main hyperparameters are: 6 samples per device, 2 gradient accumulation steps, a learning rate of 5×10^{-5} with a lineal scheduler, gradient checking, and an 8-bit quantified optimizer.

5.2 Usage and Performance

To rank the samples, we first score them. For scoring the quality of a lay summary, we used the regression model to measure the quality regarding the *Relevance*, *Readability*, and *Factuality*. With those values, we compute a single score based on the harmonic mean of those three values. The harmonic mean would give better scores to summaries that simultaneously hold high quality on the three aspects. We will refer to this score as `hm-score` for clarity.

In order to measure the ranking capabilities of the regression model, we measured the Normal Discounted Cumulative Gain (NDCG) over the real `hm-score` of the score of the best summary available and the real score of the chosen summary, based on the predicted `hm-score`.

In Fig. 1, we observe the distribution of the NDCG scores when the model ranks one approach (M1 or M2) and when the model ranks a mix of both (M1+M2). It can be noticed that with M1, it has better ranking capabilities than with M2. However, in both approaches, the scores are mainly in range of $[0.95, 1.0]$, which means that most of the time, one of the best summaries is chosen. When we mix the sources, the regression model reduces its ranking capabilities, which could indicate that it would be less precise when the quality of sum-

[5] `https://huggingface.co/kiddothe2b/biomedical-longformer-large`

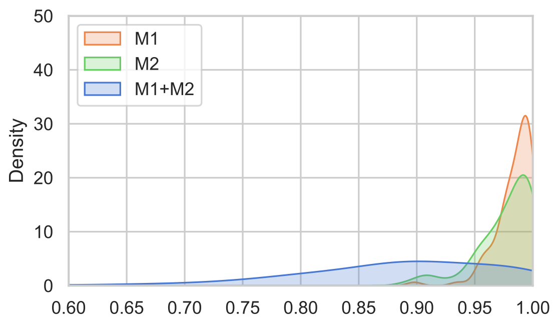


Figure 1: Distribution of the NDCG$_1$ scores obtained by the ranking model, when we consider both sources (eLife+PLOS). In `M1` and `M2`, the model ranks 10 summaries per sample; 20 summaries in `M1+M2`.

maries to choose from varies a lot.

The improvements in validation using the Ranking can be seen in Table 3 (Appendix A) can be seen for M1 M2, and M1+M2.

6 Results

For the competition, we sent a total of three submissions. ***S1*** that included lay summaries generated with M1 approach without any kind of ranking. ***S2*** that contained lay summaries generated with M1 and selected with the rank model (10 summaries per sample). Additionally, we sent a third submission (***S3***) that contained summaries from M1 and M2 and selected with the regression model (20 summaries per sample).

Table 2 shows official results for the test partition for the three submissions. It can be noticed that S2 provided the best results. Compared to S1, S2 increased the performance thanks to the ranking model. However, if the summarization model can not generate a wider variety of proposals, the ranking model will not help too much. Regarding S3, which includes the M1 and M2 summaries, we notice a lower quality of the final selection. Nevertheless, this submission increases the *Factuality* aspect, which could be attributed to the fact that M2 manages more information, reducing the factuality errors. Finally, regarding the relative performance (RP), our solution obtained more than 90% of performance in most of the scores, compared to the best overall submission. Further improvements need to be made, especially in the readability aspect.

	S1	**S2**	**S3**	**RP(%)**
Relevance				
↑ ROUGE-1	47.99	**48.15**	48.01	98.39
↑ ROUGE-2	13.61	**13.66**	13.60	87.06
↑ ROUGE-L	42.90	**43.09**	43.06	94.07
↑ BERTScore	85.94	**85.95**	85.91	99.05
Readability				
↓ FKGL	13.64	**13.61**	13.65	86.33
↓ DCRS	10.89	**10.86**	10.90	86.00
↓ CLI	14.71	**14.66**	14.70	91.13
↑ LENS	47.90	**48.02**	33.42	90.96
Factuality				
↑ AlignScore	78.37	78.21	**78.72**	97.71
↑ SummaC	60.91	60.66	**61.37**	82.67
`hm-score`	48.68	**48.69**	46.59	90.08

Table 2: Official results comparison for test partition for the three submissions (S1, S2, S3), and relative performance (RP) of S2 compared to the best overall system in the competition (UIUC_BioNLP). Bold values are the best values for each score. The up arrow (↑) indicates that the value of the score correlates positively with the quality of the lay summary, and the down arrow (↓) negatively. The `hm-score` is also included, which is not part of the official results.

7 Discussions

The results presented in Section 6 raise the benefits and constraints that must be taken into account when combining generation models with ranking models to choose which text will be presented to the end user.

Regarding the benefits, they are evident. With the ranking models, we can enhance the quality of the summaries presented to the user even though we use the same automatic summarization models. We use the ranking model to choose those summaries that obtained the best ranking scores since those texts will have better quality compared to other summaries generated by the same models. This selection should boost the overall performance of the system in most cases.

In relation to the constraints. The ranking model does not generate summaries or make texts better; it just rates summaries generated by the summarization models, and we select the best summaries based on those scores. Therefore, if summarization models have a bad performance and/or we can not provide enough variety to choose from, the benefits will be diminished. For this reason, we should combine the ranking models with summarization models that can complement each other depending on the text to summarize and offer variety in the generated summaries.

8 Conclusions

In this work, we have presented our contribution to the BioLaySumm 2024 shared task of the 23rd BioNLP Workshop. We used LED models to allow adding more text in the model input. Although we started from the same pre-trained model, different fine-tuned models were trained for the two sources of the competition: eLife and PLOS. Two different approaches were followed, one with just the technical summary as input, and another with additional text beside the technical summary. Our preliminary evaluation showed that the first approach performed better, but the second should be developed further since the larger input context improved the *Factuality* aspect. An additional contribution of our approach is the use of a regression-based ranking model that helped to boost the quality of the final summary by choosing the promising one from a set of summaries generated by the models. The model that obtained the best results in the competition was the one that combined the first approach and the ranking model.

Limitations

The data augmentation followed in this work to obtain the dataset for training the dataset is attached to the inner behavior of pre-trained LLMs. Those could present biases or limitations that we have not studied or detected. This could lead to limitations in the diversity and quality of the dataset, which could be inherited by the regression model.

Acknowledgments

This work is partially supported by MCIN/AEI/10.13039/501100011033, by the "European Union and “NextGenerationEU/MRR”, and by “ERDF A way of making Europe” under grants PDC2021-120846-C44 and PID2021-126061OB-C41. It is also partially supported by the Generalitat Valenciana under project CIPROM/2021/023, and by the Spanish Ministerio de Universidades under the grant FPU21/05288 for university teacher training.

References

Vicent Ahuir, Lluís-F. Hurtado, José Ángel González, and Encarna Segarra. 2021. Nasca and nases: Two monolingual pre-trained models for abstractive summarization in catalan and spanish. *Applied Sciences*, 11(21).

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv:2004.05150*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*.

Meri Coleman and T L Liau. 1975. A computer readability formula designed for machine scoring. *Journal of applied psychology*, 60(2):283.

Edgar Dale and Jeanne S Chall. 1948. A formula for predicting readability. *Educational research bulletin*, 27(1):11–28.

Dina Demner-Fushman, Sophia Ananiadou, Mako Miwa, Kirk Roberts, and Jun-ichi Tsujii, editors. 2024. *The 23nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*. Association for Computational Linguistics, Bangkok, Thailand.

Tomas Goldsack, Zheheng Luo, Qianqian Xie, Carolina Scarton, Matthew Shardlow, Sophia Ananiadou, and Chenghua Lin. 2023. Overview of the biolaysumm 2023 shared task on lay summarization of biomedical research articles. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 468–477, Toronto, Canada. Association for Computational Linguistics.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yue Guo, Wei Qiu, Yizhong Wang, and Trevor Cohen. 2021. Automated lay language summarization of biomedical scientific reviews. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(1):160–168.

J. Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. *Research Branch report*, 8:75.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. LENS: A learnable evaluation metric for text simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16383–16408, Toronto, Canada. Association for Computational Linguistics.

National Center for Biotechnology Information (NCBI). 2024. Pubmed: A resource by the national center for biotechnology information. `https://pubmed.ncbi.nlm.nih.gov/`.

A.s. Poornash, Atharva Deshmukh, Archit Sharma, and Sriparna Saha. 2023. APTSumm at BioLaySumm task 1: Biomedical breakdown, improving readability by relevancy based selection. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 579–585, Toronto, Canada. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Areg Mikael Sarvazyan, José Ángel González, and Marc Franco-Salvador. 2024. Textmachina: Seamless generation of machine-generated text datasets. *Preprint*, arXiv:2401.03946.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Guan Wang, Sijie Cheng, Xianyuan Zhan, Xiangang Li, Sen Song, and Yang Liu. 2023. Openchat: Advancing open-source language models with mixed-quality data. *arXiv preprint arXiv:2309.11235*.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Doug Burdick, Darrin Eide, Kathryn Funk, Yannis Katsis, Rodney Michael Kinney, Yunyao Li, Ziyang Liu, William Merrill, Paul Mooney, Dewey A. Murdick, Devvret Rishi, Jerry Sheehan, Zhihong Shen, Brandon Stilson, Alex D. Wade, Kuansan Wang, Nancy Xin Ru Wang, Christopher Wilhelm, Boya Xie, Douglas M. Raymond, Daniel S. Weld, Oren Etzioni, and Sebastian Kohlmeier. 2020. CORD-19: The COVID-19 open research dataset. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin

Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Results in Evaluation (`val` partition)

	M1	**M2**	**M1R**	**M2R**	**AR**
Relevance score	48.23	45.02	**48.28**	45.26	47.26
↑ ROUGE-1	48.88	44.06	**48.97**	44.44	47.60
↑ ROUGE-2	14.52	11.20	**14.54**	11.38	13.30
↑ ROUGE-L	43.60	40.39	**43.68**	40.77	42.70
↑ BERTScore	85.90	84.41	**85.91**	84.44	85.42
Readability score	38.51	28.16	**38.64**	28.49	37.74
↓ FKGL	13.67	15.03	13.66	14.89	**13.58**
↓ DCRS	10.85	11.69	**10.82**	11.61	10.85
↓ CLI	14.47	15.53	14.43	15.43	**14.22**
↑ LENS	49.00	23.87	**49.11**	23.61	44.20
Factuality score	68.49	**81.35**	68.16	80.96	70.37
↑ AlignScore	77.00	86.65	76.64	**85.67**	77.36
↑ SummaC	59.97	76.04	59.68	**76.25**	63.38
`hm-score`	48.94	42.85	**48.97**	43.14	48.49

Table 3: Results comparison for validation partition for the two approaches without using ranking (M1 and M2), with ranking (M1R, M2R), and M1+M2 ranked (AR). Bold values are the best values achieved for each score. The up arrow (↑) indicates that the value of the score correlates positively with the quality of the lay summary, and the down arrow (↓) negatively.

Table 3 shows the results of the two model types when one summary is requested (columns M1 and M2). Or, when 10 summaries are requested per sample, rank with our ranking model and select the top-ranked summary for each sample (columns M1+R and M2+R).

B *Relevance*, *Readability* and *Factuality* scores.

We defined *Relevance* as the average of the following scores: ROUGE-1, ROUGE-2, ROUGE-L and BERTScore. *Factuality* is the average values of AlignScore and SummaC scores.

For defining *Readability*, we start first defining the function *Clamp and Complement (CC)*:

$$CC_f^z(x) = \frac{z - f(x)|_{[0,z]}}{z} \tag{1}$$

Eq. (1) shows that, given a function f, an integer number $z > 0$, and sample x. The sample x is evaluated with f. Then, the score is clamped in a range from $[0, z]$, complemented, and normalized.

Therefore, we define *Readability* as follows:

$$\begin{aligned} Readability(x) = (& \\ & CC_{FKGL}^{20}(x) + \\ & CC_{DCRS}^{20}(x) + \\ & CC_{CLI}^{20}(x) + \\ & \frac{LENS(x)}{100} \\ &) \cdot \frac{1}{4} \end{aligned} \tag{2}$$

Eq. (2), shows that *Readability* is defined as the average of the following four scores: FKGL, DCRS, CLI, and LENS. For the three first scores (FKGL, DCRS, and CLI), the values below 20 are clamped since we consider that 20 is already a really high readability level for lay summarization purposes. Additionally, values are complemented and normalized when needed.

BioLay_AK_SS at BioLaySumm: Domain Adaptation by Two-Stage Fine-Tuning of Large Language Models used for Biomedical Lay Summary Generation

Akanksha Karotia, Seba Susan
Department of Information Technology, Delhi Technological University, Delhi, India
akankshakarotia@gmail.com, seba_406@yahoo.in

Abstract

Lay summarization is essential but challenging, as it simplifies scientific information for non-experts and keeps them updated with the latest scientific knowledge. In our participation in the Shared Task: Lay Summarization of Biomedical Research Articles @ BioNLP Workshop (Goldsack et al., 2024), ACL 2024, we conducted a comprehensive evaluation on abstractive summarization of biomedical literature using Large Language Models (LLMs) and assessed the performance using ten metrics across three categories: relevance, readability, and factuality, using eLife and PLOS datasets provided by the organizers. We developed a two-stage framework for lay summarization of biomedical scientific articles. In the first stage, we generated summaries using BART and PEGASUS LLMs by fine-tuning them on the given datasets. In the second stage, we combined the generated summaries and input them to BioBART, and then fine-tuned it on the same datasets. Our findings show that combining general and domain-specific LLMs enhances performance.

1 Introduction

In today's era, a lot of research is being conducted in the field of biomedical science, resulting in a huge amount of biomedical literature. The vast scientific knowledge poses a challenge for healthcare professionals, researchers, and the non-expert public in staying informed about advancements in the biomedical domain (Bishop et al., 2022; Karotia and Susan, 2023). Making the information accessible and understandable, regardless of their background knowledge, is difficult. Manually summarizing long scientific articles requires too much domain-oriented knowledge, effort, and time, especially for lay summarization. First, summarizing and then transforming the summarized information for non-experts is impractical. This problem can be tackled by designing lay summarization systems that bridge the gap between non-experts and experts by modifying intricate scientific knowledge into a clear and condensed form with increased readability. This step will increase scientific literacy and enable decision-making for experts and non-experts.

This study's contributions include:

- In the first phase of the model, BART (Lewis et al., 2020) and PEGASUS (Zhang et al., 2020) general domain LLMs were used. These LLMs were fine-tuned on eLife and PLOS datasets for the summarization task.
- The outputs from both LLMs are combined, and sentences are deduplicated to eliminate redundant data and enhance diversity and inclusivity.
- In the second phase, the deduplicated data was sent to the BioBART (Yuan et al., 2022) LLM, which is pre-trained on biomedical datasets and further fine-tuned by the authors on the dataset made accessible by the challenge organizers.
- Performance evaluation and analyses are done for relevance, readability, and factuality metrics.

2 Related Work

Recent studies have showcased the significant potential of large language models (LLMs) in natural language generation tasks. In addition, the first version of the BioLaySum (Goldsack et al., 2023) illustrated the effectiveness of utilizing LLMs in both summary formation (Turbitt et al., 2023) and data augmentation (Sim et al., 2023). LLMs exhibit proficiency in perceiving complex relationship patterns due to their training on diverse large-scale datasets across various tasks (Karotia and Susan, 2022).

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 762–768
August 16, 2024. ©2024 Association for Computational Linguistics

(Liu et al., 2023) utilized two different LLMs, BART and PEGASUS, for the BioLaySum task at ACL 2023, respectively focusing on the eLife and PLOS datasets, aiming to optimize memory usage. (Phan et al., 2023) processed long documents using an effective framework comprising of BioBART and a factorized energy-centric method. (Turbitt et al., 2023) performed comprehensive experiments with general and domain-specific GPT models using zero-shot and few-shot methods. (Al-Hussaini et al., 2023) utilized T5 and BART LLMs with an attention mechanism for information selection and further applied a zero-shot method for language simplification. (Reddy et al., 2023) employed BART to generate summaries by incorporating sentence labels, which significantly improved results. (Chen et al., 2023) used various models for each submission, including PRIMERA, PEGASUS, and BART-LongFormer.

3 Methodology

In recent times, Large Language Models such as BioBART pre-trained on biomedical corpora have achieved enhanced performance for biomedical natural language generation tasks. However, the readability of the generated summaries needs to be improved from the perspective of a non-expert audience. To achieve this aim, a two-stage framework is designed in this work, as shown in Figure 1, to generate lay summaries for complex and lengthy scientific research articles, targeting non-expert audiences. In the first phase of the framework, BART and PEGASUS general-purpose LLMs are selected for summary generation by fine-tuning them on the challenge datasets: eLife and PLOS. Both models performed well on validation and test sets, indicating their capability to generate high-quality summaries due to their pre-training on multiple tasks and large datasets. As these transformers have limited input lengths, the first 1024 tokens are used for training because these LLMs are resource-intensive and time-consuming. Specifically, training with starting information proves to be more efficient, as studies indicate that important information is typically presented at the beginning and end of research articles (Cai et al., 2022). BART and PEGASUS are transformer-based models that excel in text generation and are specifically designed for abstractive summarization. In the initial phase of our approach, we fine-tune these models using article-lay summary pairs from the eLife and PLOS datasets separately. This process involves setting the hyperparameters specified in Table 1, as discussed in Section 4. After fine-tuning, the summaries generated by both models are merged. The aggregated text is further processed to handle redundant information, which is eliminated through sentence deduplication. This results in diverse and non-redundant data, making it suitable to be input into the second phase of the framework for further fine-tuning.

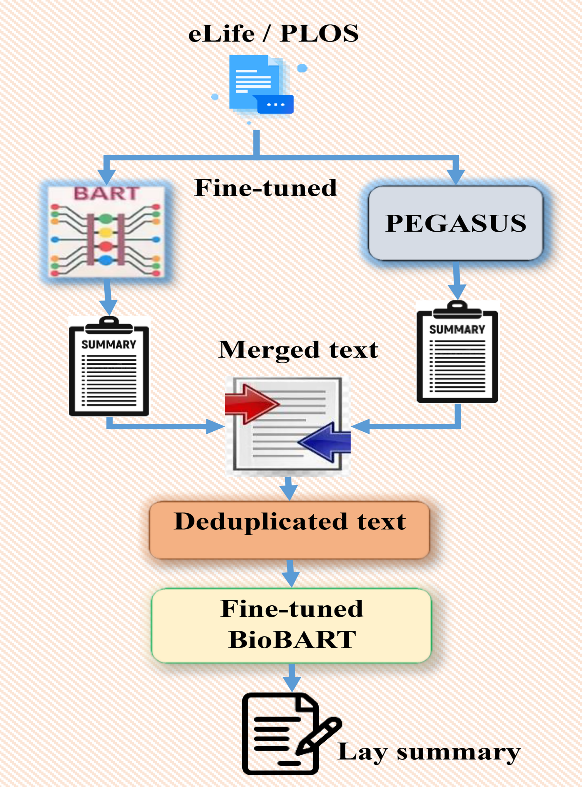


Figure 1: Proposed framework.

The authors observed two cases of redundancy after summary generation in the first phase, leading to the need for deduplication of sentences to ensure non-redundant information for the second phase. First, identical sentences are present within the same summary generated by the models. Second, the summaries generated by both models have identical sentences. This issue does not arise for all samples. But even in a few cases, it is important to ensure that non-redundant information is used for further processing to generate a quality summary.

Algorithm 1 outlines the steps for deduplicating sentences in aggregated text. This process removes identical sentences while considering case sensitivity (lower-case), ensuring non-redundant information. In the second and final phase of the framework, the deduplicated text for a correspond-

Algorithm 1: Sentence Deduplication

```
Data: $C_{txt}$ (Aggregated text)
Result: $D_{txt}$ (Deduplicated text)
$S_{tokenized} \leftarrow$ sentence_tokenize($C_{txt}$) ;
$N_{non\_duplicate\_sentences} \leftarrow$ empty list to
  store non-duplicate sentences;
$Duplicate_{ssentences} \leftarrow$ empty list to store
  duplicate sentences;
for each s in $S_{tokenized}$ do
    if s is not in $N_{non\_duplicate\_sentences}$
      then
        Append s to
          $N_{non\_duplicate\_sentences}$;
    else
        Append s to $Duplicates_{sentences}$;
    end
end
$D_{txt} \leftarrow$
  Concatenate($N_{non\_duplicate\_sentences}$);
$D_{txt}$
```

ing document is considered salient information that guides the domain-specific LLM to generate more accurate summaries. For this phase, the BioBART LLM (Yuan et al., 2022) is selected for fine-tuning, as this model is pre-trained on a vast amount of biomedical datasets and showed promising results in the first edition of the BioLaySum task. In place of the original article, the deduplicated lay summary generated in the first phase is used as input for fine-tuning BIOBART in the second phase of the proposed model. The summary obtained after fine-tuning with BioBART results in improved performance on both datasets.

Hyperparameter	**Values**
Batch size	16
Learning rate	5e-5
Early stopping	3
Max input length	1024
Min and max target length (eLife)	350, 512
Min and max target length (PLOS)	180, 200
No. of beams	4
Penalty	2

Table 1: Hyperparameter settings for all the baseline methods and the proposed model.

4 Experimental Setup

The experiments in this study are conducted on the Google Colab Pro Plus platform, using an NVIDIA A100 GPU with 40 GB of GPU RAM for training and inferencing. Appendix A provides insights into the datasets used, while Table 1 details the hyperparameter settings for all the models.

4.1 Datasets

The organizers provided the two biomedical datasets, eLife (Goldsack et al., 2022) and The Public Library of Science (PLOS) (Goldsack et al., 2022), used in this study, across which all the models were trained and evaluated. Appendix A shows the detailed dataset statistics.

4.2 Baseline and Hyperparameter Settings

[1]PEGASUS (Zhang et al., 2020): This model is pre-trained on the C4 and HugeNews datasets for abstractive summarization by utilizing the gap sentence ratio methodology, and stochastic sampling was employed for key sentence identification. It is fine-tuned on eLife and PLOS with 10 and 8 epochs, respectively.

[2]BART (Lewis et al., 2020): This model was pre-trained for language generation and translation tasks in English, and fine-tuned on the CNN/DM dataset, specifically for summarization purposes. It is fine-tuned for 13 epochs on eLife and 12 epochs on PLOS.

[3]T5-small (Raffel et al., 2020): This model, a smaller version of T5, is pre-trained on the C4 dataset for varied tasks that include paraphrasing and natural language generation. It is fine-tuned with 15 epochs on eLife and 11 epochs on the PLOS dataset.

[4]BIOBART (Yuan et al., 2022): BioBART has efficiently adapted the BART framework for generative tasks specifically tailored to the biomedical domain. The model is pre-trained on several language generation tasks, including: 1) A medical dialogue system, where the objective is to emulate a human doctor communicating with real patients, trained using the CovidDialog dataset. 2) Abstractive summarization on the iCliniq, HealthCareMagic, and MeQSum datasets. 3) Entity linking on the MedMentions, BC5CDR, and AskAPatients datasets. 4) Named entity recognition on

[1] https://huggingface.co/google/pegasus-cnn_dailymail
[2] https://huggingface.co/facebook/bart-large-cnn
[3] https://huggingface.co/google-t5/t5-small
[4] https://huggingface.co/GanjinZero/biobart-large

the ShARe13 and CADEC datasets. This model is fine-tuned with 7 epochs on eLife and 5 epochs on the PLOS dataset. The proposed model is also fine-tuned for 12 epochs using the same parameters listed in Table 1.

4.3 Evaluation metrics

Various metrics have been employed to evaluate the model's performance, categorized into three main aspects: relevance, readability, and factuality.

Relevance: Four metrics are employed to assess the relevance aspect: ROUGE-1, ROUGE-2, ROUGE-L (Lin, 2004), and BERTScore (Zhang et al.). Higher scores on these metrics indicate better performance.

Readability: The Flesch-Kincaid Grade Level (FKGL), Dale-Chall Readability Score (DCRS), Coleman-Liau Index (CLI), and LENS are used as readability metrics. Lower scores for FKGL, DCRS, and CLI indicate better readability, whereas higher scores for LENS are considered.

Factuality: AlignScore and SummaC are the metrics evaluated to measure the factual accuracy of the generated summaries. Higher scores on these metrics indicate higher quality in terms of factuality.

4.4 Results and Discussion

The performance of the baseline methods and the system proposed in this study is shown in Table 2 and Table 3 for the validation and test sets of the eLife and PLOS datasets. All performances are evaluated using the script provided by the organizers before the deadline for the validation sets of both datasets. The baselines are evaluated on the Codabench platform for the test set, but only the proposed model is evaluated after the challenge's deadline on the test set.

As shown in Table 2, the proposed model achieves better performance for relevance metrics with the best scores of ROUGE-1 (0.4681), ROUGE-2 (0.131), ROUGE-L (0.4475), and BERTScore (0.8404). It also attains the best scores for CLI (10.4805) and LENS (63.4962) metrics on the validation set of eLife. Meanwhile, the validation set of PLOS shows significantly improved performance for the readability metrics FKGL (14.1426), CLI (15.0911), and LENS (52.9523) compared to the listed baselines. Additionally, the BERTScore relevance metric performed well, scoring 0.858.

As observed from Table 3, the proposed model demonstrates superior performance in relevance metrics, achieving the best scores for ROUGE-1 (0.4635), ROUGE-2 (0.1228), ROUGE-L (0.4428), and BERTScore (0.8411). It also achieves top scores for CLI (10.9776) and LENS (65.7387) metrics on the eLife test set. In the PLOS test set, the model significantly improves readability metrics, with FKGL (13.8401), CLI (15.1084), and LENS (52.9811) scores surpassing those of the baselines. Additionally, the relevance metrics for the PLOS test set show notable improvements, with ROUGE-1 (0.4396), ROUGE-L (0.3988), and a strong BERTScore of 0.8578.

5 Conclusion and Future Scope

A two-stage fine-tuning framework combining general-purpose BART and PEGASUS LLMs with the biomedical-specific BioBART LLM showcased satisfactory performance for generating lay summaries of biomedical scientific articles. In the first phase, BART and PEGSUS were fine-tuned on the training data of eLife and PLOS datasets, while in the second phase, BioBART was fine-tuned on the merged and deduplicated lay summary generated in the first phase. All hyperparameters were set using the validation set. This framework achieved promising results for relevance and readability metrics but at the cost of marginally lower performance for factuality metrics. In the future, different combinations of domain-specific LLMs can be employed, along with language simplification techniques, to generate high-quality lay summaries for non-experts, optimizing scores for relevance, readability, and factuality. metrics. In the future, different combinations of domain-specific LLMs can be employed, along with language simplification techniques, to generate high-quality lay summaries for non-experts, optimizing scores for relevance, readability, and factuality.

6 Limitations

Adapting LLMs to new domains requires substantial fine-tuning, which may not always transfer knowledge effectively across the target domain. In the proposed model, fine-tuning of LLMs in two consecutive stages results in high computational cost and time. A significant problem is the length limitation associated with these LLMs. Although important information often resides at the beginning of scientific articles, the need to restrict input

Model	eLife									
	ROUGE-1	ROUGE-2	ROUGE-L	BERTSCORE	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
T5	0.386	0.0964	0.3742	0.8211	9.232	**6.5566**	11.7579	24.1684	0.3513	**0.7091**
BART	0.4442	0.1086	0.4145	0.8357	13.5909	10.169	13.5704	43.9951	**0.7752**	0.7033
PEGASUS	0.4068	0.1006	0.3919	0.8291	**9.5149**	6.5603	10.8472	42.6606	0.717	0.6209
BIOBART	0.4325	0.109	0.3669	0.8277	19.1071	9.7971	13.1439	27.72	0.5935	0.5149
Ours	**0.4681**	**0.131**	**0.4475**	**0.8404**	10.2465	7.8174	**10.4805**	**63.4962**	0.6264	0.5263
Model	**PLOS**									
	ROUGE-1	ROUGE-2	ROUGE-L	BERTSCORE	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
T5	0.3402	0.0973	0.3113	0.8304	15.469	**9.2281**	16.4606	35.0077	0.8612	0.7057
BART	0.4483	0.1456	0.4053	0.8565	14.3844	11.8589	15.7053	49.3006	0.8766	0.8281
PEGASUS	**0.455**	**0.1561**	**0.4123**	0.8579	14.6704	11.5939	16.2672	49.6905	0.8055	**0.8736**
BIOBART	0.4323	0.1479	0.3893	0.85	14.2208	12.07	15.9739	51.6555	**0.8991**	0.8396
Ours	0.4525	0.1458	0.4109	**0.858**	**14.1426**	11.4252	**15.0911**	**52.9523**	0.801	0.7152

Table 2: Results on the validation sets of eLife and PLOS datasets.

Model	eLife									
	ROUGE-1	ROUGE-2	ROUGE-L	BERTSCORE	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
T5	0.2614	0.0453	0.2404	0.8015	16.0014	7.6351	16.4143	26.2303	**0.9157**	0.691
BART	0.4479	0.1054	0.4169	0.8385	13.9387	10.3756	14.3257	49.2375	0.8131	**0.7296**
PEGASUS	0.3987	0.096	0.383	0.8295	**9.8937**	**6.5202**	11.1935	44.0906	0.7398	0.6401
BIOBART	0.4343	0.1043	0.3589	0.8308	20.131	9.9165	13.7818	30.8928	0.638	0.5341
Ours	**0.4635**	**0.1228**	**0.4428**	**0.8411**	10.3915	7.7965	**10.9776**	**65.7387**	0.6409	0.5443
Model	**PLOS**									
	ROUGE-1	ROUGE-2	ROUGE-L	BERTSCORE	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
T5	0.3316	0.0995	0.3034	0.8324	14.8563	**9.1379**	16.4704	34.5145	0.8647	0.6981
BART	0.4317	0.1376	0.391	0.8556	14.4732	11.8719	15.9178	49.9099	0.8876	0.8423
PEGASUS	0.4363	**0.1439**	0.3932	0.851	14.8366	11.6496	16.473	13.8482	0.8116	**0.8695**
BIOBART	0.4248	0.142	0.3839	0.8508	14.2641	12.0685	16.338	52.5272	**0.9066**	0.8366
Ours	**0.4396**	0.1405	**0.3988**	**0.8578**	**13.8401**	11.3222	**15.1084**	**52.9811**	0.8183	0.7273

Table 3: Results on the test sets of eLife and PLOS datasets.

length and truncate the rest can lead to a potential loss of information, ultimately affecting the quality of the generated summary.

References

Irfan Al-Hussaini, Austin Wu, and Cassie Mitchell. 2023. Pathology dynamics at biolaysumm: the trade-off between readability, relevance, and factuality in lay summarization. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 592–601.

Jennifer Bishop, Qianqian Xie, and Sophia Ananiadou. 2022. Gencomparesum: a hybrid unsupervised summarization method using salience. In *Proceedings of the 21st workshop on biomedical language processing*, pages 220–240.

Xiaoyan Cai, Sen Liu, Libin Yang, Yan Lu, Jintao Zhao, Dinggang Shen, and Tianming Liu. 2022. Covidsum: A linguistically enriched scibert-based summarization model for covid-19 scientific papers. *Journal of Biomedical Informatics*, 127:103999.

Chao-Yi Chen, Jen-Hao Yang, and Lung-Hao Lee. 2023. Ncuee-nlp at biolaysumm task 2: Readability-controlled summarization of biomedical articles using the primera models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 586–591.

Tomas Goldsack, Zheheng Luo, Qianqian Xie, Carolina Scarton, Matthew Shardlow, Sophia Ananiadou, and Chenghua Lin. 2023. Overview of the biolaysumm 2023 shared task on lay summarization of biomedical research articles. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 468–477.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks, Bangkok, Thailand. Association for Computational Linguistics*.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages

10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Akanksha Karotia and Seba Susan. 2022. Pre-training meets clustering: A hybrid extractive multi-document summarization model. In *International Conference on Hybrid Intelligent Systems*, pages 532–542. Springer.

Akanksha Karotia and Seba Susan. 2023. Covsumm: an unsupervised transformer-cum-graph-based hybrid document summarization model for cord-19. *The Journal of Supercomputing*, 79(14):16328–16350.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Quancheng Liu, Xiheng Ren, and VG Vinod Vydiswaran. 2023. Lhs712ee at biolaysumm 2023: Using bart and led to summarize biomedical research articles. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 620–624.

Phuc Phan, Tri Tran, and Hai-Long Trieu. 2023. Vbd-nlp at biolaysumm task 1: Explicit and implicit key information selection for lay summarization on biomedical long documents. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 574–578.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Venkat praneeth Reddy, Pinnapu Reddy Harshavardhan Reddy, Karanam Sai Sumedh, and Raksha Sharma. 2023. Iitr at biolaysumm task 1: lay summarization of biomedical articles using transformers. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 625–628.

Mong Yuan Sim, Xiang Dai, Maciej Rybinski, and Sarvnaz Karimi. 2023. Csiro data61 team at biolaysumm task 1: Lay summarisation of biomedical research articles using generative models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 629–635.

Oisín Turbitt, Robert Bevan, and Mouhamad Aboshokor. 2023. Mdc at biolaysumm task 1: Evaluating gpt models for biomedical lay summarization. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 611–619.

Hongyi Yuan, Zheng Yuan, Ruyi Gan, Jiaxing Zhang, Yutao Xie, and Sheng Yu. 2022. Biobart: Pretraining and evaluation of a biomedical generative language model. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 97–109.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International conference on machine learning*, pages 11328–11339. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Appendix

eLife: The eLife dataset contains research papers associated with lay summaries written by domain experts. The train, validation, and test sets consist of 4346, 241, and 142 articles, respectively. The average word count for lay summaries across all data splits ranges between 382-400, while for articles, it ranges from 8900-10201. Similarly, the average sentence count for lay summaries and articles ranges between 18-19 and 382-583, respectively. The minimum and maximum word counts for the train, validation, and test sets for lay summaries are (177, 686), (234, 672), and (244, 642), respectively. For articles, the minimum and maximum word counts for the train, validation, and test sets are (324, 28696), (3408, 23048), and (2492, 16880), respectively.

PLOS: This includes research papers and their corresponding lay summaries from domain experts. The dataset is divided into training, validation, and test sets with 24773, 1376, and 142 articles, respectively. Lay summaries have an average word count between 180 and 195 across all splits, whereas the articles range from 6742 to 6754 words. The average sentence length for lay summaries is 8; for articles, it is between 298 - 311 sentences. For lay summaries, the minimum and maximum word counts are 4 and 511 for the training set, 55 and 384 for the validation set, and 16 and 293 for the test set. Articles have word count ranges of 748 to 26643 for the training set, 751 to 20423 for the validation set, and 1587 to 18477 for the test set.

Data		Elife			PLOS	
	Train	Validation	Test	Train	Validation	Test
Number of samples	4346	241	142	24773	1376	142
Avg. word count (LS)	382	390	400	195	195	180
Avg. sentence count (LS)	18	18	19	8	8	8
Max. word count (LS)	686	672	642	511	384	293
Min. word count (LS)	177	234	244	4	55	16
Avg. word count (A)	10200	10021	8909	6754	6742	6939
Avg. sentence count (A)	382	583	445	299	298	311
Max. word count (A)	28696	23048	16880	26643	20423	18477
Min. word count (A)	324	3408	2492	748	751	1587

Table 1: Detailed statistics and analysis of eLife and PLOS datasets, where LS stands for Lay Summary and A stands for Article.

Model	ROUGE-1	ROUGE-2	ROUGE-L	BERTSCORE	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
T5	0.3631	0.0969	0.3428	0.8258	12.3505	7.8924	14.1093	29.5881	0.6063	0.7074
BART	0.4463	0.1271	0.4099	0.8461	13.9877	11.014	14.6379	46.6479	**0.8259**	**0.7657**
PEGASUS	0.4309	0.1284	0.4021	0.8435	**12.0927**	**9.0771**	13.5572	46.1756	0.7613	0.7473
BIOBART	0.4324	0.1285	0.3781	0.8436	16.664	10.9336	14.5589	39.6878	0.7463	0.6773
Ours	**0.4603**	**0.1384**	**0.4292**	**0.8492**	12.1946	9.6213	**12.7858**	**58.2243**	0.7137	0.6208

Table 2: Average scores achieved for eLife and PLOS on the validation set.

Model	ROUGE-1	ROUGE-2	ROUGE-L	BERTSCORE	FKGL	DCRS	CLI	LENS	AlignScore	SummaC
T5	0.2965	0.0724	0.2719	0.817	15.4289	8.3865	16.4424	30.3724	**0.8902**	0.6946
BART	0.4398	0.1215	0.404	0.8471	14.206	11.1238	15.1218	49.5737	0.8504	**0.786**
PEGASUS	0.4175	0.12	0.3881	0.8403	12.3652	**9.0849**	13.8333	28.9694	0.7757	0.7548
BIOBART	0.4296	0.1232	0.3714	0.8458	17.1976	10.9925	15.0599	41.71	0.7723	0.6854
Ours	**0.4516**	**0.1317**	**0.4208**	**0.8495**	**12.1158**	9.5594	**13.043**	**59.3599**	0.7296	0.6358

Table 3: Average scores achieved for eLife and PLOS on the test set.

WisPerMed at BioLaySumm: Adapting Autoregressive Large Language Models for Lay Summarization of Scientific Articles

Tabea M. G. Pakull[1,2], **Hendrik Damm**[2,3], **Ahmad Idrissi-Yaghir**[2,3],
Henning Schäfer[1,2], **Peter A. Horn**[1], **and Christoph M. Friedrich**[2,3]
[1] Institute for Transfusion Medicine, University Hospital Essen,
Hufelandstraße 55, 45147 Essen, Germany
[2] Department of Computer Science, University of Applied Sciences and Arts Dortmund,
Emil-Figge-Straße 42, 44227 Dortmund, Germany
[3] Institute for Medical Informatics, Biometry and Epidemiology (IMIBE),
University Hospital Essen, Hufelandstraße 55, 45147 Essen, Germany
tabeamargaretagrace.pakull@uk-essen.de

Abstract

This paper details the efforts of the WisPerMed team in the BioLaySumm2024 Shared Task on automatic lay summarization in the biomedical domain, aimed at making scientific publications accessible to non-specialists. Large language models (LLMs), specifically the BioMistral and Llama3 models, were fine-tuned and employed to create lay summaries from complex scientific texts. The summarization performance was enhanced through various approaches, including instruction tuning, few-shot learning, and prompt variations tailored to incorporate specific context information. The experiments demonstrated that fine-tuning generally led to the best performance across most evaluated metrics. Few-shot learning notably improved the models' ability to generate relevant and factually accurate texts, particularly when using a well-crafted prompt. Additionally, a Dynamic Expert Selection (DES) mechanism to optimize the selection of text outputs based on readability and factuality metrics was developed. Out of 54 participants, the WisPerMed team reached the 4th place, measured by readability, factuality, and relevance. Determined by the overall score, our approach improved upon the baseline by ≈ 5.5 percentage points and was only ≈ 1.5 percentage points behind the first place.

1 Introduction

In the biomedical domain, scientific publications and research play a central role in communicating research findings and results. However, these documents are usually written in complex language and use terminology and technical jargon that can be challenging for lay readers or researchers from different fields to understand (Goldsack et al., 2022). In this context, lay summarization can be utilized to extract the most relevant information from the original article or publication while also providing supplementary explanations. This often entails incorporating background information that may not be contained within the article itself.

In this context, this paper presents the participation of the team WisPerMed in the BioLaySumm2024 Shared Task (Goldsack et al., 2024) on automatic lay summarization and describes the employed approaches to tackle this challenge.

Summaries generated by LLMs, as demonstrated by Zhang et al. (2024), can be of equivalent or superior quality to original references. Additionally, instruction tuning is an effective approach for enhancing performance. However, LLMs face limitations when applied to domain-specific abstractive summarization. Key challenges include the quadratic complexity of transformer-based models (Vaswani et al., 2017) concerning input text length, model hallucination, where factually incorrect text is generated, and domain shift from training to test data (Afzal et al., 2023). Similarly, studies on text simplification (Amin et al., 2023) indicate that although general-purpose LLMs are capable of effectively simplifying clinical reports, they sometimes generate factual inaccuracies and omit crucial information.

To adapt LLMs to a specific domain or task (Ling et al., 2024), it is possible to fine-tune the models, leverage few-shot learning or further pre-train the models on domain data. Examples of domain-adapted LLMs for the biomedical domain include the BioMistral (Labrak et al., 2024) and OpenBioLLM (Pal and Sankarasubbu, 2024) model series. The BioMistral models are based on the Mistral 7B Instruct v0.1 (Jiang et al., 2023) model. They are further pre-trained on the PMC Open Access Subset[1]. OpenBioLLM models are based on the Llama3 (AI@Meta, 2024) models and were

[1] https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/ Accessed: 2024-05-17

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 769–779
August 16, 2024. ©2024 Association for Computational Linguistics

adapted to the biomedical domain through fine-tuning.

2 Dataset

The dataset (Goldsack et al., 2022) of the Shared Task (Goldsack et al., 2024) contains two collections of scientific journal articles and the corresponding lay summaries, namely PLOS and eLife. PLOS and eLife also include the section headings and keywords of the article. The PLOS dataset has 24,773 examples in the training split and 1,376 examples in the validation split, whereas the eLife dataset is smaller with 4,346 examples in the training split and 241 examples in the validation split. The test split consists of 142 examples for both datasets. Lay summaries of the PLOS dataset were written by the authors of the articles and are approximately 150-200 words long, while eLife lay summaries were written by expert editors in correspondence with the authors and are about twice as long.

For the remainder of this paper, any reference to the validation or test set will include eLife and PLOS unless otherwise specified.

3 Evaluation Metrics

The generated summaries were evaluated across ten metrics that fall into the following categories: relevance, readability, and factuality. Relevance was assessed through Recall-Oriented Understudy for Gisting Evaluation (Lin, 2004) (ROUGE-1, ROUGE-2, ROUGE-L) and BERTScore (Zhang et al., 2020). ROUGE counts the overlapping n-grams in the generated texts and target lay summaries, whereas BERTScore uses contextual word embeddings to compare the semantic similarity of the two texts. Readability was evaluated using the Flesch-Kincaid Grade Level (FKGL) (Kincaid, 1975), Dale-Chall Readability Score (DCRS) (Chall and Dale, 1995), Coleman-Liau Index (CLI) (Coleman and Liau, 1975), and Learnable Evaluation Metric for Text Simplification (LENS) (Maddela et al., 2023). The FKGL uses sentence lengths and syllable count per word to estimate readability. The DCRS uses a word list to compute the occurrences of words unknown to most 4-th graders and the CLI estimates the grade level necessary to comprehend the text. The LENS metric is a learnable evaluation metric trained on datasets containing human ratings of simplifications. In this setting, LENS measures the simplification of the abstract by the generated text using the target lay summary as a reference. Factuality was assessed with AlignScore (Zha et al., 2023) and Summary Consistency (SummaC) (Laban et al., 2022). The AlignScore quantifies the degree of alignment between the facts in the summary and the scientific article, while SummaC also includes consistency.

4 Methods and Experiments

This section outlines the methodology employed in the experiments conducted on the specified dataset.

4.1 Fine-tuned Models

In this study, instruction tuning (Wei et al., 2022) was utilized to fine-tune various models. Instruction tuning refers to the process of fine-tuning language models on a collection of datasets described via instructions. BioMistral-7B-DARE (BioM) and Llama3-70B-Instruct (Llama3) were fine-tuned for one epoch utilizing Quantized Low-Rank Adaptation (QLoRA) (Dettmers et al., 2023) on the eLife and PLOS dataset individually. BioM was trained on the abstracts + lay summaries, whereas Llama3 was trained on the entire articles + lay summaries. The texts were structured using the Mistral and Llama3 instruction templates prior to the fine-tuning process. Please refer to the Appendix A, B, and C for details on prompts, parameters and licenses, respectively.

After evaluating the checkpoints of BioM on the validation set, the checkpoints with the best scores were selected for inference. For Llama3, the final checkpoints were selected. The models were given the same prompt as during fine-tuning but without the target.

4.2 Prompt Variations

Prompts can guide the LLM's content generation process without the need for fine-tuning. In the zero- and few-shot settings, different prompt variations and their effect on the evaluation metrics were examined. In the few-shot setting, example lay summaries from the training and validation set were included in the prompt when performing inference on the validation and test set, respectively. The format of these few-shot prompts is designed to emulate a preceding conversation with the model, with the included examples serving as the model's previous responses.

To choose the best few-shot examples, all examples were ranked based on their average nor-

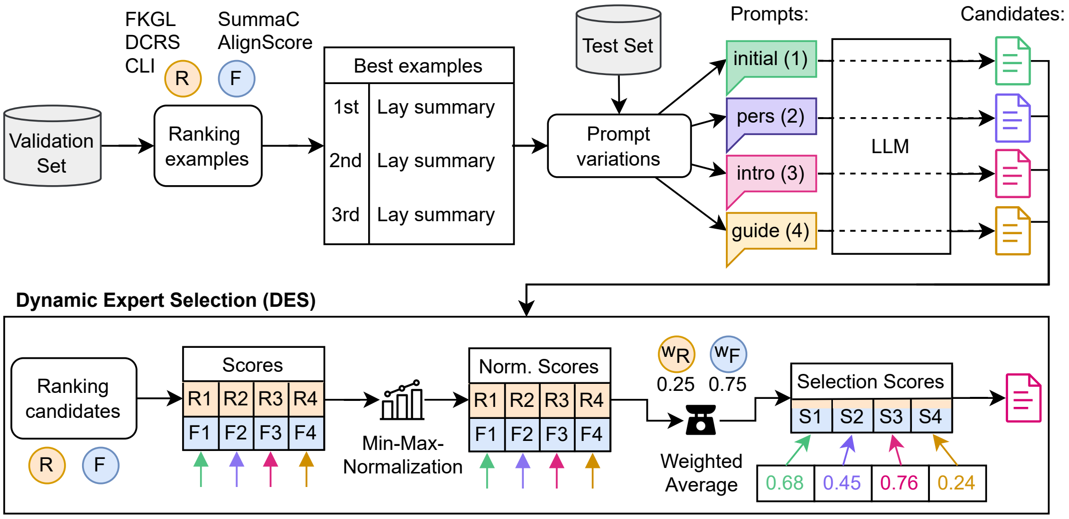


Figure 1: Workflow of the Dynamic Expert Selection (DES) mechanism in the few-shot setting using an example from the PLOS dataset. The process involves ranking examples, generating multiple summaries through various prompt variations, applying a large language model (LLM), and then normalizing and weighing the readability (R) and factuality (F) scores to rank and select the best summary based on the selection scores (S).

malized readability and factuality. The two and three highest-ranked examples were selected for the eLife and PLOS datasets, respectively.

An initial prompt was created by replicating the prompt used for inference with the fine-tuned BioM model (see Appendix A). This prompt was then tested with BioM and OpenBioLLM-70B (OpenBio) on the validation set.

Additionally, three prompt variations were created, which provide the model with different kinds of context information. It was decided that BioM would be utilized for all experiments involving these variations due to its superior performance on the validation set in the few-shot setting (see Table 3 in Appendix D). LLMs can assume different roles and adapt their vocabulary accordingly (Salewski et al., 2023), resulting in enhanced performance in tasks related to the specified role. Accordingly, the first prompt variation comprises a persona description of a science communicator (BioM$_{\text{pers}}$), instructing the model to utilize the expertise of this persona to create the lay summary based on the abstract. The model is then instructed to channel the expertise of the described persona to craft the lay summary based on the abstract. The second prompt variation is a modification of the initial prompt, incorporating the introduction to provide additional background information because associated context can improve LLM performance (Karmaker and Feng, 2023). The second prompt variation is a modification of the initial prompt, but it includes the introduction as further context for background information (BioM$_{\text{intro}}$). The third prompt variation includes the abstract and a guide on how to write a lay summary (BioM$_{\text{guide}}$), accompanied by instructions concerning the content and style of the requested summary. This method leverages the importance of clear and detailed task directives. The selection of these prompts was based on a few preliminary experiments with the model and an initial assessment of the responses. However, no comprehensive optimization was performed. The wording of all prompts can be found in Appendix A.

Due to the efficacy of few-shot learning with the initial prompt, the prompt variations were implemented in a few-shot setting on the test set.

4.3 Dynamic Expert Selection (DES)

The success of an LLM depends on factors such as the properties of the dataset, the complexity of the domain, and the design of the prompt (Ling et al., 2024). Consequently, a model may yield a more suitable lay summary when prompted in a different manner. In addition, the output quality depends upon the selection of the inference parameters (Minaee et al., 2024). In consideration of this assumption, a Dynamic Expert Selection (DES) was developed. It selects the most appropriate text from a set of candidate texts based on metrics that do not require a reference lay summary.

The mechanism uses the readability metrics FKGL, DCRS, and CLI, as well as the factuality metrics AlignScore and SummaC. These metrics are computed for each candidate text. Readability scores are multiplied by -1 so that higher scores indicate better readability. All scores are normalized using min-max normalization to range between 0 and 1, where 1 is the best and 0 is the worst. For each candidate text, an overall score is calculated by multiplying the means with different weights. Given that the target lay summaries in eLife have a higher readability than those in PLOS (Goldsack et al., 2022), the overall scores are computed with different weights for the two aspects. For eLife summaries: Readability is weighted at 0.675 and factuality at 0.325. For PLOS summaries: Readability is weighted at 0.25 and factuality at 0.75. The candidate text with the highest overall score is selected as the most suitable lay summary. The selection of the weights is based on the assumptions about the target texts and comparisons of the overall scores on the validation dataset.

This approach was applied to BioM in the few-shot setting using all prompt variants (see Figure 1) and to the fine-tuned BioM using two distinct inference parameter settings (see Appendix B).

5 Results

The results of the experiments using BioM, Llama3, and OpenBio are presented in table 1. The experiments are categorized into zero-shot learning, few-shot learning, and fine-tuning.

BioM exhibits the highest LENS score in the zero-shot setting. However, its relevance and factuality performance are the lowest. Few-shot learning resulted in enhanced performance across all metrics except for LENS. The persona prompt (BioM$_{\text{pers}}$) led to an improvement in relevance. Including the introduction in the prompt (BioM$_{\text{intro}}$) resulted in a reduction in all aspects despite the fact that the model had access to more information from the article itself. In comparison, the prompt with the guide (BioM$_{\text{guide}}$) exhibits minimal enhancements. The optimal few-shot learning for BioM occurred with the initial prompt, which achieved the highest readability and factuality in the few-shot setting, excluding the DES approach. However, OpenBio slightly underperformed with this prompt in the few-shot setting, except for the LENS score, where it performed best in this setting.

The DES used all four prompts and outperformed the baseline with improvements in factuality and readability, achieving the best results in the few-shot setting.

Fine-tuning BioM improved relevance and factuality scores, though the LENS score decreased slightly, with other readability metrics similar to the few-shot setting. The fine-tuned BioM outperformed the baseline in terms of relevance and overall quality. The DES approach improved all metrics except for a slight drop in the LENS score. In contrast, Llama3 underperformed despite being larger. It was less effective at extracting relevant information from full articles and produced lower-quality text in terms of readability, even though its LENS score was higher than BioM's. Additionally, Llama3's factuality scores decreased, leading to an overall performance drop compared to the baseline.

6 Conclusion

This paper presents the WisPerMed team's approaches to automatic lay summarization within the biomedical domain, utilizing a combination of fine-tuning, prompt variations, and Dynamic Expert Selection.

Among these approaches, fine-tuning emerged as an effective method, leading to the best performance across most metrics. This underscores the importance of task-specific training in optimizing model output for complex summarization tasks. Additionally, BioM showed strong few-shot learning capabilities, illustrating its robustness and versatility in generating accurate and relevant summaries even without extensive training. As the model adjusts to the factuality and readability of given examples, providing better examples could lead to further enhancements in these aspects.

BioM reached high factuality, even when provided solely with abstracts as input, suggesting that BioM leveraged domain-specific knowledge acquired during pre-training. This indicates that domain adaptation remains an important factor when using LLMs for lay summarization of scientific articles, as BioM outperformed the larger general model Llama3.

The four prompt variations exhibited differing effects on the evaluation metrics. BioM is adept in fulfilling the role of a science communicator (BioM$_{\text{pers}}$), as evidenced by the enhanced relevance. BioM$_{\text{intro}}$ and BioM$_{\text{guide}}$ did not significantly enhance the metrics, indicating that the increase in context was not beneficial for all texts. Without DES, a shorter prompt (BioM$_{\text{initial}}$) yielded the optimal results, suggesting that the model effectively

Expt.	R-1	R-2	R-L	BERT	FKGL	DCRS	CLI	LENS	Align	SC
Baseline	0.470	0.140	0.436	0.862	12.036	10.148	13.485	48.096	0.779	0.703
					Zero-shot Learning					
BioM	0.329	0.071	0.298	0.845	12.404	10.093	13.974	80.396	0.541	0.458
					Few-shot Learning					
BioM	0.440	0.124	0.409	**0.857**	11.287	**8.954**	**12.755**	75.744	0.728	0.604
BioM$_{\text{pers}}$	**0.442**	0.125	**0.412**	0.856	11.318	9.066	13.031	63.766	0.721	0.607
BioM$_{\text{intro}}$	0.391	0.106	0.359	0.851	12.233	9.618	13.693	76.638	0.669	0.529
BioM$_{\text{guide}}$	0.434	0.117	0.403	0.856	11.773	9.553	13.662	76.912	0.692	0.557
BioM$_{\text{DES}}$	0.439	**0.128**	0.409	0.855	**10.969**	8.993	12.819	74.025	**0.767**	**0.673**
OpenBio	0.415	0.104	0.382	0.855	11.657	9.848	13.711	**79.519**	0.731	0.558
					Fine-tuning					
BioM	0.470	**0.152**	0.442	**0.865**	11.338	8.872	13.064	51.058	0.775	0.705
BioM$_{\text{DES}}$	**0.471**	**0.152**	**0.443**	**0.865**	**11.072**	**8.862**	**12.871**	51.028	**0.782**	**0.722**
Llama3	0.418	0.108	0.391	0.856	11.622	10.628	15.080	**72.860**	0.602	0.592

Table 1: Performance metrics of experiments on the test set. The models include BioMistral-7B (BioM), Llama3-70B (Llama3), and OpenBioLLM-70B (OpenBio). The experiments are categorized into fine-tuned, zero-shot, and few-shot settings. The metrics reported are ROUGE-1 (R-1), ROUGE-2 (R-2), ROUGE-L (R-L), BERTScore (BERT), FKGL, DCRS, CLI, LENS, AlignScore (Align), and SummaC (SC). Bolded values indicate the best in each section, and underlined values the best overall performance.

comprehends the task from the provided examples. The DES mechanism further refined readability and, in particular, factuality by retrospectively selecting the best text outputs based on evaluation metrics. This highlights the potential of metric-driven selection to improve the quality of lay summaries further.

In conclusion, our study demonstrates that fine-tuning, the use of informed prompt variations, and selection mechanisms can enhance the capability of autoregressive LLMs to produce lay summaries that are factually accurate, relevant, and readily accessible to non-specialist audiences. This approach fosters broader public engagement with scientific findings, advancing the goal of making biomedical research comprehensible and accessible.

Limitations

Only four discrete prompts in combination were tested with DES, and only two sets of inference parameters were explored. This limited scope means that the findings may not fully capture the potential variability and performance of the the various models under different conditions. The weights for the Dynamic Expert Selection method were chosen based on heuristics without any formal optimization, which could impact the robustness and generalizability of the results. Another limitation is the possibility that BioM may have been previously exposed to the gold standard summaries. If this is the case, it could skew the results by artificially inflating the model's performance. These limitations indicate potential avenues for future research, including the necessity for more comprehensive prompt engineering, optimization of DES weights, and a wider range of tasks to ensure the robustness of the approach. Another potential future direction is adapting these methods for other complex domains or languages and exploring additional metrics.

Acknowledgement

The work of Tabea M. G. Pakull, Hendrik Damm, Ahmad Idrissi-Yaghir and Henning Schäfer was funded by a PhD grant from the DFG Research Training Group 2535 *Knowledge-and data-based personalisation of medicine at the point of care (WisPerMed).*

References

Anum Afzal, Juraj Vladika, Daniel Braun, and Florian Matthes. 2023. Challenges in Domain-Specific

Abstractive Summarization and How to Overcome Them. In *Proceedings of the 15th International Conference on Agents and Artificial Intelligence*, pages 682–689, Lisbon, Portugal. SCITEPRESS - Science and Technology Publications.

AI@Meta. 2024. Llama 3 Model Card.

Kanhai Amin, Pavan Khosla, Rushabh Doshi, Sophie Chheang, and Howard P. Forman. 2023. Artificial Intelligence to Improve Patient Understanding of Radiology Reports. *The Yale Journal of Biology and Medicine*, 96(3):407–417.

J. S. Chall and E. Dale. 1995. *Readability Revisited: The New Dale-Chall Readability Formula*, volume 1. Brookline Books.

Meri Coleman and T. L. Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283–284.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient Finetuning of Quantized LLMs. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023*, New Orleans, LA, USA.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the BioLaySumm 2024 Shared Task on the Lay Summarization of Biomedical Research Articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making Science Simple: Corpora for the Lay Summarisation of Scientific Literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B. *arXiv preprint*. ArXiv:2310.06825.

Shubhra Kanti Karmaker and Dongji Feng. 2023. TELeR: A General Taxonomy of LLM Prompts for Benchmarking Complex Tasks. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14197–14203, Singapore. Association for Computational Linguistics.

J.P. Kincaid. 1975. *Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel*. Research Branch report. Chief of Naval Technical Training, Naval Air Station Memphis, Millington, Tennessee.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-Visiting NLI-based Models for Inconsistency Detection in Summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. BioMistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains. *arXiv preprint*. ArXiv:2402.10373.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Workshop on Text Summarization Branches Out, Post-Conference Workshop of ACL 2004*, pages 74–81, Barcelona, Spain.

Chen Ling, Xujiang Zhao, Jiaying Lu, Chengyuan Deng, Can Zheng, Junxiang Wang, Tanmoy Chowdhury, Yun Li, Hejie Cui, Xuchao Zhang, Tianjiao Zhao, Amit Panalkar, Dhagash Mehta, Stefano Pasquali, Wei Cheng, Haoyu Wang, Yanchi Liu, Zhengzhang Chen, Haifeng Chen, Chris White, Quanquan Gu, Jian Pei, Carl Yang, and Liang Zhao. 2024. Domain Specialization as the Key to Make Large Language Models Disruptive: A Comprehensive Survey. *arXiv preprint*. ArXiv:2305.18703.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *7th International Conference on Learning Representations*, New Orleans, LA, USA. OpenReview.net.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. LENS: A Learnable Evaluation Metric for Text Simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16383–16408, Toronto, Canada. Association for Computational Linguistics.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large Language Models: A Survey. *arXiv preprint*. ArXiv:2402.06196.

Ankit Pal and Malaikannan Sankarasubbu. 2024. OpenBioLLMs: Advancing Open-Source Large Language Models for Healthcare and Life Sciences.

Leonard Salewski, Stephan Alaniz, Isabel Rio-Torto, Eric Schulz, and Zeynep Akata. 2023. In-Context Impersonation Reveals Large Language Models' Strengths and Biases. In *Advances in Neural Information Processing Systems*, volume 36, pages 72044–72057. Curran Associates, Inc.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned Language

Models are Zero-Shot Learners. In *International Conference on Learning Representations*.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating Factual Consistency with A Unified Alignment Function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. In *8th International Conference on Learning Representations*, Addis Ababa, Ethiopia. OpenReview.net.

Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori B. Hashimoto. 2024. Benchmarking Large Language Models for News Summarization. *Transactions of the Association for Computational Linguistics*, 12:39–57.

A Prompts

The prompts used in the experiments are shown in Figures 2, 3, 4, 5, and 6.

Fine-tuning/Initial Prompt with Abstract for BioM

You will be provided with the abstract of a scientific article. Your task is to write a lay summary that accurately conveys the key findings and significance of the research in non-technical language understandable to a general audience.:

Abstract of a scientific article:

[Abstract]

Lay summary for this article:

Figure 2: The prompt used for fine-tuning BioM and as the initial prompt in the zero- and few-shot settings. For fine-tuning the prompt also includes the target lay summary.

Fine-tuning/Inference Prompt with Article for LLama3

You will be provided with a scientific article. Your task is to write a lay summary that accurately conveys the key findings and significance of the research in non-technical language understandable to a general audience.:

Scientific article:

[Abstract]

Lay summary for this article:

Figure 3: The prompt used for fine-tuning Llama3. For fine-tuning the prompt also includes the target lay summary.

Persona Prompt

Meet Layla, your fantastic science communicator committed to breaking down complex research for everyone! Layla's mission is to create summaries that make scientific literature easy to understand for the general public. Before writing, Layla thoroughly reads the abstract to grasp the research goals and findings accurately. Precision is crucial for Layla; she makes sure her summaries align with the abstract's research while expanding on key points and methods. Layla ensures each summary gives a complete understanding of the findings and their importance. She offers detailed explanations and backgound information as context to aid comprehension. She highlights the main discoveries and their real-world implications, explaining study mechanisms and methods in reader-friendly language. Layla brings research to life with vivid descriptions and relatable examples, showing its impact on society. Her tone is informative yet engaging, avoiding jargon to be inclusive.

Now, let's channel Layla's expertise to craft a comprehensive lay summary for a scientific article.

Abstract of the scientific article:

[Abstract]

Layla:

Figure 4: The Persona-Prompt used in zero- and few-shot setting with BioM.

Intro Prompt

You will be provided with the abstract of a scientific article and the introduction section for background information. Your task is to write a lay summary that accurately conveys the key findings and significance of the research in non-technical language understandable to a general audience. Please ensure that your summary is mainly based on the information provided in the abstract. You may also use information from the introduction for additional context if necessary.

Introduction of the scientific article:

[Introduction]

Abstract of the scientific article:

[Abstract]

Lay summary for this article:

Figure 5: The Intro-Prompt used in zero- and few-shot setting with BioM.

Guide Prompt

You will be provided with the abstract of a scientific article. Your task is to write a lay summary that accurately conveys the key findings and significance of the research in non-technical language understandable to a general audience.

Abstract of the scientific article:

[Abstract]

Guidelines for crafting a lay summary:

- Craft a detailed summary that explains the research findings and their implications, providing thorough explanations where necessary.
- Ensure factual accuracy and alignment with the research presented in the abstract, elaborating on key points and methodologies.
- Highlight the main findings and their implications for real-world scenarios, delving into specific mechanisms or methodologies used in the study and their broader significance.
- Incorporate descriptive language to explain complex concepts.
- Maintain a balanced tone that is informative and engaging, avoiding technical jargon or overly formal language.
- Ensure the summary provides sufficient depth and context to guide the reader through the research journey and address potential questions or areas of confusion.

Your lay summary for the article:

Figure 6: The Guide-Prompt used in zero- and few-shot setting with BioM.

B Setup and Hyperparameter

Training All trainings were executed on a single Nvidia H100 80GB using the unsloth[2] framework and QLoRA (Dettmers et al., 2023). The following modules were targeted with QLoRA: “q_proj”, “k_proj”, “v_proj”, “o_proj”, “gate_proj”, “up_proj”, and “down_proj”. The QLoRA rank and alpha were both set to 16. The QLoRA dropout was set to 0. The optimization of the models was conducted using the 8-bit Adam optimizer (Loshchilov and Hutter, 2019), which was configured with a maximum learning rate of 2×10^{-4} and a weight decay factor of 0.01. The learning rate schedule included a linear decay following an initial phase consisting of five warm-up steps. Maximum sequence length was set to 4,096.

Inference For the inference process, a greedy search algorithm was employed as the decoding strategy (Minaee et al., 2024), with a configuration that allowed for the generation of up to 1024 new tokens per inference iteration.

DES The DES with the fine-tuned model used the inference parameter as described above for one candidate, and a repetition penalty of 1.1 was chosen to generate another candidate.

C Licenses

In Table 2 the Licenses as given by the owners of the Framework/Model are displayed.

Framework/Model	License
unsloth[3]	Apache License Version 2.0
BioMistral-7B-DARE[4]	Apache License Version 2.0
Llama-3-70B-I[5]	Llama 3 Community License Agreement
OpenBioLLM-70B[6]	Llama 3 Community License Agreement

Table 2: Licenses of the dataset, Framework and Models used for this Shared Task.

[2] `https://github.com/unslothai/unsloth` Accessed: 2024-05-17

[3] `https://github.com/unslothai/unsloth` Accessed: 2024-05-17

[4] `https://huggingface.co/BioMistral/BioMistral-7B-DARE` Accessed: 2024-05-17

[5] `https://huggingface.co/meta-llama/Meta-Llama-3-70B-Instruct` Accessed: 2024-05-17

[6] `https://huggingface.co/aaditya/Llama3-OpenBioLLM-70B` Accessed: 2024-05-17

D Results on the Validation Set

The results of experiments on the validation set and the reference scores of the target lay summaries and input abstracts are presented in Table 3.

Expt.	R-1	R-2	R-L	BERT	FKGL	DCRS	CLI	LENS	Align	SC
Targets	-	-	-	-	12.857	9.944	14.251	57.988	0.670	0.512
Abstracts	0.410	0.135	0.380	0.855	15.260	11.378	16.961	38.259	-	-
Zero-shot Learning										
BioM	0.332	0.070	0.301	0.844	**12.530**	10.156	**13.957**	**80.159**	0.521	0.465
BioM$_{\text{pers}}$	0.411	0.118	0.379	0.847	12.579	**10.074**	14.897	69.732	0.741	0.628
BioM$_{\text{intro}}$	0.397	0.118	0.364	0.849	13.735	10.478	14.990	68.530	0.743	0.580
BioM$_{\text{guide}}$	**0.422**	**0.123**	**0.389**	**0.851**	13.971	10.478	15.667	68.561	**0.747**	**0.593**
Few-shot Learning										
BioM	**0.440**	**0.122**	**0.411**	**0.855**	**10.875**	**8.733**	**12.359**	76.358	**0.701**	**0.596**
OpenBio	0.423	0.107	0.390	0.854	12.429	9.729	14.721	**77.961**	0.678	0.554
Fine-tuning										
BioM	0.478	0.148	0.446	0.866	11.743	9.899	13.886	56.888	0.724	0.677

Table 3: Performance metrics of experiments on the validation set. The models include BioMistral-7B (BioM), Llama3-70B (Llama3), and Llama3-OpenBioLLM-70B (OpenBio). The experiments are categorized into fine-tuned, zero-shot, and few-shot settings. The metrics reported are ROUGE-1 (R-1), ROUGE-2 (R-2), ROUGE-L (R-L), BERTScore (BERT), FKGL, DCRS, CLI, LENS, AlignScore (Align), and SummaC (SC). 'Targets' and 'Abstracts' provides benchmark scores of the target lay summaries and abstracts, respectively. Bolded values indicate the best in each section, and underlined values the best overall performance.

HULAT-UC3M at BiolaySumm: Adaptation of BioBART and Longformer models to summarizing biomedical documents

Adrián González, Paloma Martínez
Computer Science and Engineering Department
Universidad Carlos III de Madrid
100494633@alumnos.uc3m.es, pmf@inf.uc3m.es

Abstract

This article presents our submission to the BioLaySumm 2024 shared task: Lay Summarization of Biomedical Research Articles. The objective of this task is to generate summaries that are simplified in a concise and less technical way, in order to facilitate comprehension by non-experts users. A pre-trained BioBART model was employed to fine-tune the articles from the two journals, thereby generating two models, one for each journal. The submission achieved the 12th best ranking in the task, attaining a meritorious first place in the Relevance ROUGE-1 metric.

1 Introduction

In the context of the rapidly expanding quantity and complexity of biomedical literature, the ability to effectively and accurately summarise documents has become crucial for researchers and healthcare professionals. In this regard, Natural Language Processing (NLP) technologies have emerged as promising tools to address this need. The objective of BioLaySumm 2024 shared task (Goldsack et al., 2024) is the simplification of biomedical research articles playing a vital role in making information more comprehensible to non-experts thus enabling a wider audience to understand and use medical information effectively.

Concerning generating summaries, there are a number of different approaches that can be employed. One such approach is the extractive model, which involves selecting the most important sentences from the original text and incorporating them directly into the summary. These models were the first to emerge and the most widely used until the abstractive models came onto the scene. These models have the capacity to comprehend the content of the input text and generate summaries that may include new sentences and expressions that are not present in the original text (Nallapati et al., 2017)(Widyassari et al., 2022). The first paper to describe an abstractive summarisation model was (Cohan et al., 2018), and from that moment on, they began to gain greater relevance and were used more frequently than the extractive models. In this paper, we will employ abstractive models.

In our participation in the BioLaySumm 2024 shared task, we utilise existing large language models (LLMs), such as Bio-BART (Yuan et al., 2022), which is a biomedical variant of the BART model (Lewis et al., 2020), and Longformer Encoder-Decoder (LED) (Beltagy et al., 2020), to train our models for the generation of summaries from the provided articles. The summaries generated by the various models were evaluated in accordance with the metrics provided by the organisers. (ROUGE(1,2 and L) (Lin, 2004), BERTScore (Zhang et al., 2020), FKGL (Kincaid et al., 1975), DCRS (Chall and Dale, 1995), CLI (Coleman and Liau, 1975), LENS (Maddela et al., 2023), AlignScore (Zha et al., 2023) and SummaC (Laban et al., 2021)). The experiment that yielded the most favourable results was the one that used the BioBART pre-trained model. This model was used to train two models, one for each of the journals from which the articles were obtained. These models were used to generate the abstracts for each journal.

This release achieved excellent results in the Relevance metric of the shared task, with the highest score in the ROUGE-1 metric. However, the Readability and Factuality metrics yielded less impressive outcomes, resulting in a final ranking of 12th place. Nevertheless, this remains a satisfactory performance, as it places the team in the top half of the table of all participants.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 780–785
August 16, 2024. ©2024 Association for Computational Linguistics

PLOS Dataset			
data	train	validation	test
size	24733	1376	142
avg-length	6754.09	6741.48	6939.28
min-length	748	751	1587
max-length	26643	20423	18477

Table 1: Data statistics of PLOS dataset. Size corresponds to the number of articles present in the dataset. The min-length and max-length values correspond to the minimum and maximum length of the words in the dataset. Finally, avg-length corresponds to the average word length of all texts in the dataset.

eLife Dataset			
data	train	validation	test
size	4346	241	142
avg-length	10200.27	10031.25	8909.15
min-length	324	3408	2492
max-length	28696	23048	16880

Table 2: Data statistics of eLife dataset.

2 Method

2.1 Dataset

In order to participate in the BioLaySumm 2024 shared task, all participants are provided with two datasets containing biomedical research articles, the expert abstract, the name of the article sections and finally the keywords of each article. The first dataset contains approximately 25,000 articles from the Public Library of Science (PLOS), while the second dataset contains approximately 5,000 articles from the journal eLIfe. Details of the dataset are provided in (Goldsack et al., 2022).

In the tables 1 and 2 we can see the different statistics of the two journals (PLOS and eLife), in them we can see for each split its length of texts, the average number of words in each split, as well as the maximum and minimum length. The average length of articles varies depending on the journal to which they belong. For example, the average length of articles in the eLife journal is 10,200 words, while the average length of articles in PLOS is 6,754 words. In addition, there are notable differences in the length of the abstracts. The average length of an eLife abstract is twice that of a PLOS abstract, at 382 words versus 194, respectively.

2.2 Models

In order to generate the summaries, a number of approaches were tested, with two Transformer models being employed: Longformer (LED) and BioBART.

Longformer

Upon examination of the datasets in the previous study, it became evident that the articles were relatively lengthy. This prompted the decision to utilise a model that could process a substantial number of tokens as input. Consequently, the Longformer model, specifically the LED (Longformer Encoder - Decoder) variant, was selected (Beltagy et al., 2020). This model follows a sequence-to-sequence architecture (seq2seq) and is based on Transformer-base models. However, these are limited to short input sequences due to the exponential growth in computational complexity with the length of the inputs. Longformer models address this issue by introducing a mechanism whereby the complexity grows linearly in relation to the inputs. For the experiments, the pre-trained model *allenai/led-base-16384*[1] was utilised, which is capable of supporting inputs of up to 16,000 tokens. This is feasible due to the fact that it was initiated from a BART-base model. However, the BART model is only capable of processing texts up to 1,000 tokens. Consequently, the embedding matrix from the BART-base was copied and replicated 16 times in order to enable the Longformer model to process texts up to 16,000 tokens.

Bio-BART

Given the nature of this biomedical article summarisation and simplification task, it was deemed appropriate to utilise a model that has been pre-trained in this specific domain. Consequently, the BioBART model was employed (Yuan et al., 2022), as it has already demonstrated its efficacy in tasks of a similar nature and was employed in last year's task such as in (Phan et al., 2023). This model is based on a base BART model that has been pre-trained on a corpus of biomedical texts, rendering it an optimal choice for biomedical tasks. The model for the experiments is the pre-training *GanjinZero/biobart-v2-large*[2].

[1] https://huggingface.co/allenai/led-base-16384

[2] https://huggingface.co/GanjinZero/biobart-v2-base

3 Experiments

3.1 Evaluation Measures

Submissions for the shared task are evaluated according to three distinct criteria: relevance, readability and factuality.

- The relevance measure assesses the extent to which the generated abstract contains the key information from the original article. Four metrics will be employed to evaluate this: ROUGE-1 ↑, ROUGE-2↑, ROUGE-L ↑ (Lin, 2004) and BERTScore↑ (Zhang et al., 2020).
- Readability is a measure of the readability of the generated abstract, with the objective of ensuring that it is as understandable as possible for humans. In evaluating the readability of the abstract, four metrics are employed: Flesch-Kincaid Grade Level (FKGL) ↓ (Kincaid et al., 1975), Dale-Chall Readability Score (DCRS)↓ (Chall and Dale, 1995), Coleman-Liau Index (CLI) ↓ (Coleman and Liau, 1975) and LENS ↑ (Maddela et al., 2023).
- Factuality is the extent to which the generated summary is accurate and based on verifiable facts. For this, two metrics will be employed: AlignScore ↑ (Zha et al., 2023) and SummaC ↑ (Laban et al., 2021).

The objective of the relevance and factuality measures is to maximise the metrics, while in relevance we seek to minimise them, except for the LENS metric, which, like the previous ones, we seek to maximise.

3.2 Experiments

Three distinct experiments were conducted utilising the two previously trained models.

Longformer

The pre-trained *allenai/led-base-16384* model is employed in the experiments, which is capable of supporting inputs of up to 16,384 tokens. In this experiment, a single model will be trained on the texts of the two journals, and the summaries will be generated from the same model. Consequently, the training of the model employs the texts of the two journals. Despite the maximum input capacity of the model being 16,384 tokens, the texts are limited to those below 12,000 words due to identified constraints. Nevertheless, the training is based on more than 20,000 texts.

Bio-BART

The experiment employs the *GanjinZero/biobart-v2-large* pre-training model, which is a biomedical pre-training model. However, as a bart model, it has an input limitation of 1024 tokens. Consequently, for the training process, the complete dataset is utilised, with only the initial tokens of each text being employed. This approach allows for the retention of the initial tokens, which are then used for the training process. The information retained is the abstract, which has an average length of 300 tokens plus the beginning of the introduction. The average number of tokens in these two fields is 1080, demonstrating that by utilising these two sections, we are able to retain a substantial amount of information. In contrast to the aforementioned experiment with the LED model, two distinct training sets will be employed in this instance. One will comprise articles from the PLOS journal, while the other will comprise articles from the eLife journal. This approach will result in the generation of two independent models, each of which will produce summaries of the articles in their respective test sets. The fine-tuning process will utilise both complete datasets.

Longformer + Bio-BART

Finally, in order to enhance the outcomes of the preceding experiments, we opted to integrate the two models in order to retain the most advantageous aspects of each. This integration will allow us to leverage the capacity of the LED model to process voluminous text inputs while simultaneously capitalising on the BioBART model's aptitude for biomedical simplifications. As with the BioBART model, in this experiment we will utilise two independent models, one for PLOS journal and one for eLife.

In order to achieve this, the Longformer model is first employed. The input for this model is the full articles, and the output is between 700 and 800 words, which is more than double the average length of the final summaries to be delivered. Once the first summaries have been generated by the Longformer model, they are used as input to the BioBART models, which generate the final summaries.

	Relevance				**Readability**				**Factuality**	
	ROUGE-1↑	ROUGE-2↑	ROUGE-L↑	BERTScore	FKGL↓	DCRS↓	CLI↓	LENS↑	AlignScore↑	SummaC↑
Best Score	0.487	0.156	0.454	0.867	10.459	6.760	11.044	81.205	0.930	0.902
LED	0.411	0.113	0.386	0.846	13.592	**8.810**	14.966	27.749	**0.753**	0.652
BioBART	**0.487**	**0.147**	**0.452**	**0.862**	**12.710**	10.433	14.080	49.344	0.667	**0.670**
LED + BioBART	0.456	0.131	0.426	0.857	13.025	9.605	**13.360**	**52.124**	0.580	0.540

Table 3: The results of the three experiments (LED, BioBART, LED + BioBART) are presented alongside a comparison with the best results obtained in each metric in the competition.

		Relevance				**Readability**				**Factuality**	
		ROUGE-1↑	ROUGE-2↑	ROUGE-L↑	BERTScore	FKGL↓	DCRS↓	CLI↓	LENS↑	AlignScore↑	SummaC↑
LED	Average	0.411	0.113	0.386	0.846	13.592	8.810	14.966	27.749	0.753	0.652
	PLOS	0.421	0.142	0.393	0.855	13.389	8.956	14.850	29.155	0.784	0.701
	eLife	0.400	0.084	0.379	0.837	13.794	**8.665**	15.082	26.343	0.723	0.604
BioBART	Average	0.487	0.147	0.452	0.862	12.710	10.433	14.080	49.344	0.667	0.670
	PLOS	0.465	**0.155**	0.425	**0.863**	14.566	11.936	16.550	34.079	**0.791**	**0.827**
	eLife	**0.509**	0.138	**0.479**	0.861	10.854	8.930	**11.610**	**64.609**	0.542	0.514
LED + BioBART	Average	0.456	0.131	0.426	0.857	13.025	9.605	13.360	52.124	0.580	0.540
	PLOS	0.426	0.134	0.392	0.857	15.231	10.365	15.034	41.056	0.651	0.597
	eLife	0.487	0.127	0.459	0.856	**10.820**	8.846	11.685	63.192	0.509	0.540

Table 4: The results of the metrics in the PLOS and eLife journals for each of the three experiments are presented below.

3.3 Environment Parameters

All experiments were conducted on a Tesla T4 GPU, with a series of hyperparameters set, including a learning rate of 2e-5, a batch size of 4, and two epochs.

4 Results and discussions

The table 3 presents the outcomes of the experiments, displayed in the context of the various metrics. Furthermore, an additional row has been included, in which the best value for each metric within the competition is presented. Table 4 presents the results obtained for each journal, allowing for a more detailed analysis. Upon examination of the results, the following observations can be made.

The first of these observations is that our BioBART value in the ROUGE-1 metric is the best value in the competition. In addition to this excellent result in this metric, we can also see that in the other relevance metrics we also obtain very good results, being very close to the best results. Furthermore, an analysis of the results by journal reveals that there are minimal differences between the texts of the two groups. The journal PLOS outperforms the other texts in two metrics (ROUGE-2 and BERTScore), while eLife excels in two others (ROUGE-1 and ROUGE-L). This indicates that the model generates summaries that retain a substantial amount of relevant information. In the experiment in which we combined LED and BioBART, we also obtained very good results, which suggests that these results are due to the BioBART model.

Conversely, an analysis of the Readability metrics reveals that the optimal outcome is achieved when the two models are combined. However, when the Dalle-Chall Readability Score (DCRS) metric is considered, the LED model exhibits significantly superior performance. Furthermore, this metric presents an intriguing phenomenon: the results in the BioBART model are quite poor, with a score of 1.5 points above our best result. This is a significant drawback for the model in terms of its final score. In contrast to the previous observation regarding relevance, the texts of the journal eLife obtain much better results than those of the journal PLOS.

With regard to the Factuality metrics, the BioBART model yielded the most favourable results, with the exception of the eLife journal, where the outcomes were considerably less favourable. Consequently, the average score was reduced, resulting in the LED model, which is more balanced, achieving better results in the AlignScore metric.

The findings of this study indicate that while the information is well-maintained, as evidenced by the relevance metrics. The PLOS journal articles contain more accurate information but are more challenging to comprehend. This discrepancy may be attributed to the smaller abstracts (175-220 words), which may have a detrimental impact on the readability metrics.

The BioBART model is the most effective in terms of relevance, outperforming all the metrics in this category thanks to its specific biomedical training. Although the combined Longformer+BioBART model improves readability, it loses accuracy due to the double simplification of the content. On the other hand, the Longformer model, although it obtained good results in some metrics, did not stand out in any of them; this could be an effect of having trained a single model with the texts of the two journals.

4.1 Selection of approach

Following the completion of the three experiments and the analysis of the results obtained from the various metrics, it was determined that the most optimal approach would be to utilise the BioBART model, as it yielded the most favourable outcomes in six out of the ten metrics, with at least one in each of the three categories.

5 Conclusions

This paper presents our participation in the BioLaySumm 2024 shared task, which aimed to generate lay summaries of large biomedical documents. In this task, we trained two different models (LED and BioBART) from which we generated three different experiments. Upon completion of the task, we observed that the best results were obtained by training two BioBART models (one for the PLOS journal articles and another for the eLife articles). This is our final submission to the competition, which resulted in a 12th-place finish. Our performance was particularly noteworthy in the ROUGE-1 metric, where we achieved first place, as well as in the Relevance metrics.

As future work, we would have liked to experiment with other models that we found interesting, particularly trained with medical data, such as medical mT5 (García-Ferrero et al., 2024). With respect to the models we have presented, we would like to continue working with them to improve the results in the Readability and Factuality metrics, in which we have not obtained such good results. We would like to study what happened in generating not adequate summaries by conducting an analysis of errors. We believe that managing specific medical terminology would help to generate more lay-oriented medical terms to ascertain the efficacy of the keyword translation from the original text to the summary. In the event that this process is not executed correctly, due to the inherent complexity of the keywords, an external dataset comprising words from the biomedical field and a translation into simpler expressions could be employed as a preprocessing step for the texts prior to training. See the open-access and collaborative (OAC) consumer health vocabulary[3] (CHV) as an example of a medical lay-oriented vocabulary.

Limitations

Our best result is obtained by using a BioBART model, which restricts the input of words to a maximum length of 1024 tokens. This represents the initial and most significant limitation encountered, given that the dataset comprises lengthy texts. Consequently, this limitation precludes the training of models with all available information, which would result in enhanced outcomes. Another limitation identified was the use of the Tesla T4 GPU. The extensive training data required for this device resulted in lengthy training times, which impeded the development of the models.

Acknowledgments

This work was supported by ACCESS2MEET project (PID2020-116527RB-I0) supported by MCIN AEI/10.13039/501100011033/.

References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *Preprint*, arXiv:2004.05150.

J.S. Chall and E. Dale. 1995. *Readability Revisited: The New Dale-Chall Readability Formula*. Brookline Books.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *Preprint*, arXiv:1804.05685.

Meri Coleman and T. L. Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283–284.

Iker García-Ferrero, Rodrigo Agerri, Aitziber Atutxa Salazar, Elena Cabrio, Iker de la Iglesia, Alberto Lavelli, Bernardo Magnini, Benjamin Molinet, Johana Ramirez-Romero, German Rigau, Jose Maria Villa-Gonzalez, Serena Villata, and Andrea Zaninello. 2024. Medical mt5: An open-source multilingual text-to-text llm for the medical domain. *Preprint*, arXiv:2404.07613.

[3] https://biomedinfo.smhs.gwu.edu/chv-files

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Peter Kincaid, Robert P. Fishburne, Richard L. Rogers, and Brad S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2021. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *Preprint*, arXiv:2111.09525.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. Lens: A learnable evaluation metric for text simplification. *Preprint*, arXiv:2212.09739.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Phuc Phan, Tri Tran, and Hai-Long Trieu. 2023. VBD-NLP at BioLaySumm task 1: Explicit and implicit key information selection for lay summarization on biomedical long documents. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 574–578, Toronto, Canada. Association for Computational Linguistics.

Adhika Pramita Widyassari, Supriadi Rustad, Guruh Fajar Shidik, Edi Noersasongko, Abdul Syukur, Affandy Affandy, and De Rosal Ignatius Moses Setiadi. 2022. Review of automatic text summarization techniques methods. *Journal of King Saud University - Computer and Information Sciences*, 34(4):1029–1046.

Hongyi Yuan, Zheng Yuan, Ruyi Gan, Jiaxing Zhang, Yutao Xie, and Sheng Yu. 2022. Biobart: Pretraining and evaluation of a biomedical generative language model. *Preprint*, arXiv:2204.03905.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *Preprint*, arXiv:2305.16739.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

Saama Technologies at BioLaySumm: Abstract based fine-tuned models with LoRA

Hwanmun Kim, Kamal Raj Kanakarajan, Malaikannan Sankarasubbu
Saama Technologies
{hwan.kim, kamal.raj, malaikannan.sankarasubbu}@saama.com

Abstract

Lay summarization of biomedical research articles is a challenging problem due to their use of technical terms and background knowledge requirements, despite the potential benefits of these research articles to the public. We worked on this problem as participating in BioLaySumm 2024. We experimented with various fine-tuning approaches to generate better lay summaries for biomedical research articles. After several experiments, we built a LoRA model with unsupervised fine-tuning based on the abstracts of the given articles, followed by a post-processing unit to take off repeated sentences. Our model was ranked 3rd overall in the BioLaySumm 2024 leaderboard. We analyzed the different approaches we experimented with and suggested several ideas to improve our model further.

1 Introduction

While many academic publications in the biomedical field can potentially benefit a wide readership including many non-experts, their accessibility is often limited by their use of technical terms and relatively sophisticated expressions. Therefore the summarization of biomedical research articles is an interesting and important task that can benefit the general public, and BioLaySumm 2024 (Goldsack et al., 2024) aims to solve this question by adopting techniques of NLP. BioLaySumm asks participants to suggest models that summarize the biomedical articles based on the PLOS and eLife datasets (Goldsack et al., 2022) composed of original research articles and lay summaries written by experts.

In this paper, we explain our approaches to the BioLaySumm 2024 in detail. To generate better lay summaries, we experimented with multiple fine-tuning approaches with LoRA based on the abstract part of the biomedical research papers. As a result of a series of experimentations, we concluded that our best-performing model is the unsupervised fine-tuned model with LoRA followed by a post-processing unit that chops off repeated sentences in the raw predictions. At the end of the competition, our model was ranked 3rd overall in BioLaySumm 2024 leaderboard.

2 Background

2.1 Task description

In BioLaySumm 2024, participants are expected to generate lay summaries for the research articles in the test set made from PLOS and eLife journals. For the development of summarization systems, PLOS (eLife) dataset provides 24773 (4346) articles for the train split and 1376 (241) articles for the validation split. For both PLOS and eLife datasets, the test split is composed of 142 articles. For each data point, the whole article including the abstract is provided along with the keywords and article id. For the train splits and the validation splits, ground-truth lay summaries targeted for non-experts are provided. These summaries are written by authors (PLOS) or expert editors (eLife). Participants can submit summaries generated from either individual models for each dataset or a unified model for both datasets. The qualities of submitted summaries are evaluated in three criteria: relevance, readability, and factuality. Each criterion is composed of multiple automatic metrics:

- **Relevance**: ROUGE (1,2, and L) (Lin, 2004), BERTScore (Zhang et al., 2020)
- **Readability**: Flesch-Kincaid Grade Level (FKGL) (Kincaid et al., 1975), Dale-Chall Readability Score (DCRS) (Dale and Chall, 1948), Coleman-Liau Index (CLI) (Coleman and Liau, 1975), LENS (Maddela et al., 2023)
- **Factuality**: AlignScore (Zha et al., 2023), SummaC (Laban et al., 2022)

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 786–792
August 16, 2024. ©2024 Association for Computational Linguistics

These metrics are calculated through the BioLaySumm 2024 evaluation script[1]. For each metric, the average score over the entire prediction is reported. The goal of competition is to minimize FKGL, DCRS, and CLI and maximize all other metrics.

2.2 Related works

While automatic text summarization has long been the subject of interest for its wide applicability in various domains (El-Kassas et al., 2021; Allahyari et al., 2017), the advent of large language models (LLMs) has innovated the field drastically (Chang et al., 2024; Zhang et al., 2024; G et al., 2024).

As a subfield of text summarization, automatic lay summarization of biomedical literature obtained further attention for its close relationship with health literacy (Guo et al., 2021). Since most biomedical research articles assume readers are familiar with the scientific concepts and domain-specific languages of the field, it is important to measure and evaluate the readability of the generated summaries as well (Guo et al., 2021; Goldsack et al., 2022). On the other hand, fact-checking the lay summaries has been important as the use of LLMs becomes popular since LLMs are known to often experience hallucinations that generate misinformed texts (Zhang et al., 2023).

In this context, BioLaySumm provides a meaningful challenge where both the readability and factuality of summaries are evaluated (Goldsack et al., 2023, 2024). While various approaches were used for last year's competition (Goldsack et al., 2023), the most successful approaches include few-shot prompting on GPT models (Turbitt et al., 2023), fine-tuning on FLAN-T5 models (Sim et al., 2023), and factorized energy-based model trained on BioBart model (Phan et al., 2023).

3 System overview

To find the best-performing system for BioLaySumm 2024, we experimented with several different systems based on the abstracts of the research articles. In this section, we introduce the systems we experimented including the system we submitted to the leaderboard of BioLaySumm 2024. Throughout all experiments, we used eLife (PLOS) training data only for model training or prompting to generate summaries for eLife (PLOS) validation/test data.

3.1 Submitted system: Unsupervised fine-tuned LoRA model

The system we submitted for the competition is the unsupervised fine-tuned LoRA model. Due to the context-size limitation of most LLMs, it is nearly impossible to fit the entire articles into the inputs for the LLMs. Instead, inspired by the system (Turbitt et al., 2023) which took 1st place in the last year's competition (Goldsack et al., 2023), we only appended the abstract and the lay summary for the inputs to the model (Template 1). We used the entire input text for our training phase while we only used the input text just before the lay summary starts for the text generation. For parameter-efficient training, we adopted low-rank adaptation (LoRA) (Hu et al., 2021) for our training.

```
### Provide a lay summary of the following
    research abstract.

Abstract: In temperate climates , winter deaths
    exceed summer ones . However , there is
    limited information on the timing and the
    relative magnitudes of maximum and minimum
    mortality , (...)
Lay summary: In the USA , more deaths happen in
    the winter than the summer . But when deaths
     occur varies greatly by (...)
```

Template 1: Input text for unsupervised fine-tuning. The bold-faced text is the part used for the text generation as well.

While examining the generated summaries, we found that our fine-tuned model tends to repeat identical sentences rather than ending the summary. To regulate this, we post-processed our summary to chop off the redundant sentences. See appendix A For the details of the post-processing.

3.2 Other approaches

3.2.1 Baseline: zero-shot and few-shot prompting

While we use some form of fine-tuning in all the other approaches, we set a few-shot prompting system as our baseline following the best-performing system from the previous year's competition (Turbitt et al., 2023). While we adopted this abstract-based few-shot approach from the last year's competition, we randomly sampled 6 examples from the train set instead of hand-picked 3 examples used in the last year. We listed 6 abstract-summary pairs out of these sampled examples. See appendix B

[1] https://github.com/TGoldsack1/BioLaySumm2024-evaluation_scripts

for the sample prompt we used. Also, to provide a baseline that indicates the bare ability of the LLM we use, we tested zero-shot prompts where the same template was used as the few-shot prompts but with no examples listed.

3.2.2 Supervised fine-tuning with LoRA

Since the input text used in unsupervised fine-tuning in Section 3.1 trains not only the styles of lay summaries but also the styles of the original abstracts to the model, the quality of generated summaries may be affected by these abstracts in unwanted ways. To prevent this, we experimented with supervised fine-tuning. In particular, we treated the content of the lay summary as the label and the rest of the input text as the context by excluding input text tokens from the calculation of the loss function. To make this 'label' to be automatically detected after tokenization, we slightly changed the format of input text from Template 1 (see Appendix C).

3.2.3 Direct preference optimization on the fine-tuned model

Since our fine-tuning approaches only use abstract-summary pairs, it does not see the full contents of the research article during the training. Therefore the generated summaries may struggle with the factuality criterion. To mitigate this problem, we experimented with direct preference optimization (DPO) (Rafailov et al., 2024). DPO trains the human preference on a language model by providing pairs of similar samples where the relative preference within each pair is labeled (preferred sample vs rejected sample). To provide these relative preference labels, we generated summaries on randomly sampled 1000 articles in the train set using the unsupervised fine-tuned model and calculated factuality metrics (AlignScore, SummaC) on both the ground-truth lay summary and the generated summary. After comparing the average of the calculated factuality metrics within each pair, we label the summary with the higher score as the preferred sample and the summary with the lower score as the rejected sample. This DPO training is performed on top of the unsupervised fine-tuned model in Section 3.1.

4 Experimental setup

4.1 Hardware

All our experiments performed on a 4× Quadra RTX 8000 (48GB VRAM) card.

4.2 Text generation

We used `mistral-7B-instruct-v0.2` throughout all experiments. For both the few-shot approach and the fine-tuned approach, text generation is performed through vLLM[2] (Kwon et al., 2023) for faster experimentation. We set the temperature to 0 for all text generation.

4.3 Fine-tuning experiments with LoRA

For both unsupervised and supervised fine-tuning experiments, we utilized libraries from Huggingface (Transformers, PEFT[3], TRL[4]). We used AdamW optimizer (Loshchilov and Hutter, 2017) to optimize cross-entropy loss with label smoothing (Pereyra et al., 2017). Experimented hyperparameters are available in Appendix D.

4.4 Direct preference optimization experiments

For DPO experiments, we utilized Axolotl library[5]. We used the sequence size of 4096, the batch size 8, and the learning rate 1.0×10^{-5} with a linear scheduler over 3 epochs.

5 Results

5.1 Experiment results

We report the results of all our experiments in Table 1. Averages of result 7 and result 8 are the scores submitted to the leaderboard of BioLaySumm2024, and our model is ranked 2nd in relevance, 16th in readability, 18th in factuality, and 3rd in average scores of all categories out of 55 participants (Goldsack et al., 2024). Overall, our model delivered decent summaries in all 3 evaluation criteria while particularly successful in the relevance criterion.

5.2 Analysis on approaches

5.2.1 Baseline approaches: zero-shot and few-shot prompting

We set the zero-shot and few-shot prompting system as our baseline following the most successful approach last year (Turbitt et al., 2023). When comparing the baseline results from others in Table 1 (result 1, 3 vs. result 5, 9∼13 and result 2, 4 vs. result 6), fine-tuning approaches outperform zero-shot or few-shot prompting in relevance. For readability, fine-tuning is superior for the eLife

[2] https://github.com/vllm-project/vllm
[3] https://github.com/huggingface/peft
[4] https://github.com/huggingface/trl
[5] https://github.com/OpenAccess-AI-Collective/axolotl

#	Apporach	Dataset	Relevance				Readability				Factuality	
			R-1	R-2	R-L	BS	FKGL	DCRS	CLI	LENS	AS	SC
1	Baseline: Zero-shot	eLife, V	0.335	0.089	0.308	0.843	13.34	10.44	14.90	**74.90**	0.680	0.503
2	Baseline: Zero-shot	PLOS, V	0.442	0.128	0.400	0.861	13.50	**10.46**	14.90	**75.27**	0.680	0.527
3	Baseline: Few-shot	eLife, V	0.466	0.128	0.437	0.859	11.63	9.33	12.80	69.60	**0.711**	0.506
4	Baseline: Few-shot	PLOS, V	0.465	0.150	0.427	0.867	**12.86**	11.00	**13.97**	65.59	0.838	0.684
5	Unsup. FT	eLife, V	**0.497**	**0.150**	**0.477**	**0.865**	8.70	7.46	10.41	64.24	0.623	0.531
6	Unsup. FT	PLOS, V	**0.500**	**0.191**	**0.464**	**0.873**	14.16	10.67	15.52	45.25	**0.941**	**0.873**
7	Unsup. FT	eLife, T	0.477	0.133	0.456	0.863	8.52	7.36	10.42	62.31	0.601	**0.553**
8	Unsup. FT	PLOS, T	0.480	0.176	0.443	0.871	14.20	10.84	15.89	41.48	0.956	0.901
9	Sup. FT	eLife, V	0.488	0.143	0.467	0.863	10.86	7.90	10.13	63.58	0.607	0.510
10	Unsup. FT + DPO	eLife, V	0.487	0.144	0.467	0.863	8.43	7.34	10.40	63.40	0.630	0.537
11	Unsup. FT, no PP	eLife, V	0.493	0.149	0.473	**0.865**	8.72	7.40	10.40	63.97	0.621	0.531
12	Sup. FT, no PP	eLife, V	0.478	0.141	0.457	0.862	10.89	7.69	**10.10**	62.68	0.602	0.509
13	Unsup. FT + DPO, no PP	eLife, V	0.473	0.141	0.453	0.863	**8.41**	**7.10**	10.38	62.29	0.624	0.536

Table 1: All experiment results. The # column indicates the experiment result number. The approach column describes the components of the approach used for that experiment, such as zero-shot, few-shot, unsupervised fine-tuning (unsup. FT), supervised fine-tuning (sup. FT), direct preference optimization (DPO), or post-processing (PP). The dataset column indicates the dataset and the split (T for test, V for validation). For further clarification, we highlighted the results for the PLOS dataset with blue shades. Here we report all the 10 metrics used for BioLaySumm 2024: ROUGE-1 (R-1), ROUGE-2 (R-2), ROUGE-L (R-L), BERTScore (BS), Flesch-Kincaid Grade Level (FKGL), Dale-Chall Readability Score (DCRS), Coleman-Liau Index (CLI), LENS, AlignScore (AS), and SummaC (SC). Bold-faced numbers indicate the best scores we obtained on the validation split of each dataset.

dataset (except for the LENS score) while the opposite is true for the PLOS dataset. This might be related to the worse readability of PLOS summaries that the authors write themselves. On the other hand, fine-tuning approaches yield higher factuality scores for the PLOS dataset while giving worse AlignScore and better SummaC scores for the eLife dataset. These contrastive patterns in readability and factuality among different datasets might indicate that readability and factuality are in a trade-off relationship, as simplified summaries may deliver less accurate information.

5.2.2 Unsupervised vs. supervised fine-tuning

By comparing the unsupervised fine-tuning experiments (results 5, 11) with the supervised fine-tuning experiments (results 9, 12) in Table 1, we find that unsupervised fine-tuning outperforms supervised fine-tuning in all metrics except CLI. Despite our expectation of supervised fine-tuning performing better in the readability scores from not learning the patterns in the abstracts, the supervised fine-tuning was not superior in the readability neither. Detailed investigations on the reasons for this difference between the supervised and the unsupervised fine-tuning would be a good subject for the future research.

5.2.3 Direct preference optimization

When comparing the results of DPO experiments (results 10, 13) with the results of their fine-tuned model before DPO training (results 5, 11) in Table 1, we observe that DPO training gives better factuality scores as expected, as well as improved readability scores except for LENS. Yet, DPO training makes relevance scores worse at the same time, as its training process suggests some ground truth summaries as rejected samples.

5.2.4 Post-processing

To investigate the effect of the post-processing unit, we evaluated predictions with no post-processing (results 11, 12, 13 in Table 1). The comparison with the results of post-processed summaries (results 5, 9, 10) shows that post-processed summaries are superior to non-processed summaries in both relevance and factuality. Regarding the readability, the effect of the post-processing unit is mixed, where the post-processing improves LENS while it worsens DCRS and CLI. For FKGL, the effect is not consistent over different experiments.

6 Conclusion

As we participated in BioLaySumm 2024, we experimented with different fine-tuning approaches with LoRA to generate summaries based on the given abstract of a biomedical research article. In particular, we explored unsupervised fine-tuning, supervised fine-tuning, and direct preference optimization, and we concluded that our best-performing model is the unsupervised fine-tuned model with post-processing to chop off repeated sentences. Our model achieved 3rd place overall in the leaderboard of BioLaySumm 2024. While our model was successful, it would be interesting to extend our approach to a variety of larger LLMs or to adopt other schemes to utilize the full article of the research paper instead of the abstracts. Potential future researches on analysis on different fine-tuning methodologies and benchmarking on evaluation criteria beyond the current challenge may deepen the understanding of our approach.

7 Limitations

Due to our limited resources, we only experimented with a single type of relatively small open-sourced model. Due to the limited context size of the model we used, our exploration of methods to utilize full research articles was limited to DPO which interacts with the full articles only through the factuality scores.

It is also worthwhile to mention that our approach was more successful in the relevance than other than two other evaluation criteria. This might be related with the fact that summaries more readable than the suggested golden summary might score less in the BERTScore. It would be interesting subject for the future researches to see how our approach performs in other summary evaluation criteria beyond the current challenge.

References

Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assefi, Saeid Safaei, Elizabeth D. Trippe, Juan B. Gutierrez, and Krys Kochut. 2017. Text summarization techniques: A brief survey. *International Journal of Advanced Computer Science and Applications*, 8(10).

Yapei Chang, Kyle Lo, Tanya Goyal, and Mohit Iyyer. 2024. Booookscore: A systematic exploration of book-length summarization in the era of llms. *Preprint*, arXiv:2310.00785.

Meri Coleman and Ta Lin Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283.

Edgar Dale and Jeanne Sternlicht Chall. 1948. *A formula for predicting readablility*. Bureau of Educational Research, Ohio State University.

Wafaa S. El-Kassas, Cherif R. Salama, Ahmed A. Rafea, and Hoda K. Mohamed. 2021. Automatic text summarization: A comprehensive survey. *Expert Systems with Applications*, 165:113679.

Bharathi Mohan G, Prasanna Kumar R, Vifert Jenuben Daniel V, Archanaa. N, Mohammed Faheem, Suwin Kumar. J.D. T, and Kousihik. K. 2024. Comparative evaluation of large language models for abstractive summarization. In *2024 14th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, pages 59–64.

Tomas Goldsack, Zheheng Luo, Qianqian Xie, Carolina Scarton, Matthew Shardlow, Sophia Ananiadou, and Chenghua Lin. 2023. Overview of the biolaysumm 2023 shared task on lay summarization of biomedical research articles. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 468–477, Toronto, Canada. Association for Computational Linguistics.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yue Guo, Wei Qiu, Yizhong Wang, and Trevor Cohen. 2021. Automated lay language summarization of biomedical scientific reviews. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(1):160–168.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. LENS: A learnable evaluation metric for text simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16383–16408, Toronto, Canada. Association for Computational Linguistics.

Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*.

Phuc Phan, Tri Tran, and Hai-Long Trieu. 2023. VBD-NLP at BioLaySumm task 1: Explicit and implicit key information selection for lay summarization on biomedical long documents. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 574–578, Toronto, Canada. Association for Computational Linguistics.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Mong Yuan Sim, Xiang Dai, Maciej Rybinski, and Sarvnaz Karimi. 2023. CSIRO Data61 team at BioLaySumm task 1: Lay summarisation of biomedical research articles using generative models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 629–635, Toronto, Canada. Association for Computational Linguistics.

Oisín Turbitt, Robert Bevan, and Mouhamad Aboshokor. 2023. MDC at BioLaySumm task 1: Evaluating GPT models for biomedical lay summarization. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 611–619, Toronto, Canada. Association for Computational Linguistics.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori B. Hashimoto. 2024. Benchmarking Large Language Models for News Summarization. *Transactions of the Association for Computational Linguistics*, 12:39–57.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *Preprint*, arXiv:2309.01219.

A Details of post-processing

In the raw predictions of fine-tuned models, we observed that identical sentences are repeated without completing the paragraph in a small fraction of the generated summaries. To mitigate this, we introduced the post-processing unit to chop off the repeated sentences from the prediction. To do this, we first split the prediction into a sequence of sentences. Then we examine these sentences from the beginning of the sequence and drop the rest of the sequence when the given sentence has appeared before during the examination.

We split the prediction into sentences based on the appearance of sentence-ending punctuation marks like period (".") or question mark ("?"). Yet, there are some exceptions we had to handle in this process:

- If punctuation is in the middle of parentheses, does not end the sentence there.
- If a period is part of a URL address, which is specified by the beginning sequences ("www" or "http") and the ending sequences ("com", "edu", "gov", or "org"), then do not end the sentence at that period.
- If a period is part of commonly used abbreviations in academic writing ("et al .", "vs .", and "e . g ."), do not end the sentence at that period.
- If the previous word of a period is a single letter English alphabet, do not end the sentence there, since it is likely a part of a phrase for a subsection or abbreviation of names (ex: "a.1", "c. elegans", "George R. R. Martin").
- If a period is surrounded by Arabic numerals, do not end the sentence since it is likely a part of a floating number.

B Few-shot prompt for the baseline system

```
### Provide a lay summary of the following
    research abstract.

Abstract: The role of the cellular
    microenvironment in enabling metazoan tissue
     genesis remains obscure . Ctenophora has
    recently emerged as (...)
Lay summary: The emergence of the diversity of
    multicellular animals involved cells joining
     together to form tissues and organs . The
    glue that (...)

Abstract: To evolve and to be maintained ,
    seasonal migration , despite its risks , has
     to yield fitness benefits compared with
    year-round residency . Empirical data
    supporting this (...)
Lay summary: Winter is one of the most
    challenging seasons for many animals . Cold
    temperatures , bad weather , short days ,
    long nights and a shortage of food can
    impose (...)

Abstract: The adaptive prokaryotic immune system
     CRISPR-Cas provides RNA-mediated protection
     from invading genetic elements . The
    fundamental basis of the system is (...)
Lay summary: In most animals , the adaptive
    immune system creates specialized cells that
     adapt to efficiently fight off any viruses
    or other pathogens that have invaded . (...)

Abstract: Adipose tissue is crucial for the
    maintenance of energy and metabolic
    homeostasis and its deregulation can lead to
     obesity and type II diabetes ( T2D ) .
    (...)
Lay summary: Obesity is a growing public health
    concern around the world , and can lead to
    the development of type 2 diabetes , heart
    disease and cancer . (...)

Abstract: The roles played by cortical
    inhibitory neurons in experience-dependent
    plasticity are not well understood . Here we
     evaluate (...)
Lay summary: What we see or fail to see through
    our eyes leaves a lasting impression by
    changing the strength of connections between
     (...)

Abstract: Numerous studies have established
    important roles for microRNAs ( miRNAs ) in
    regulating gene expression . Here , we
    report that miRNAs also serve as (...)
Lay summary: To produce a protein from a gene ,
    the sequence of the gene must be transcribed
     to produce a molecule of messenger RNA (
    mRNA ) . (...)

Abstract: Midbrain dopamine neurons have been
    proposed to signal reward prediction errors
    as defined in temporal difference ( TD )
    learning algorithms. (...)
Lay summary:
```

Template 2: Sample few-shot prompt used for our baseline system. The 6 examples listed here are the actual examples we used for the eLife articles.

C Input text for supervised fine-tuning

```
### Provide a lay summary of the following
    research abstract.

### Abstract: In temperate climates , winter
    deaths exceed summer ones . However , there
    is limited information on the timing and the
     relative magnitudes of maximum and minimum
    mortality , (...)

### Lay summary: In the USA , more deaths happen
     in the winter than the summer . But when
    deaths occur varies greatly by (...)
```

Template 3: Input text for supervised fine-tuning. The bold-faced text is the context and the rest of the text is the label.

D Fine-tuning hyperparameters

Hyperparameter	Values
Epochs	**3**, 5
Batch size	**8**
Sequence size	2048, **4096**
Learning rate (LR)	1.0E-5, **2.0E-5**
LR scheduler	**Linear**
LoRA r	**8**
LoRA α	**16**

Table 2: Hyperparameters we investigated in the fine-tuning experiments. Hyperparameters in bold are what we used for the submitted model.

AUTH at BioLaySumm 2024: Bringing Scientific Content to Kids

Loukritia Stefanou and **Tatiana Passali** and **Grigorios Tsoumakas**
School of Informatics, Aristotle University of Thessaloniki
{loukritia,scpassali,greg}@csd.auth.gr

Abstract

The BioLaySumm 2024 shared task at the ACL 2024 BioNLP workshop aims to transform biomedical research articles into lay summaries suitable for a broad audience, including children. We utilize the BioBART model, designed for the biomedical sector, to convert complex scientific data into clear, concise summaries. Our dataset, which includes a range of scientific abstracts, enables us to address the diverse information needs of our audience. This focus ensures that our summaries are accessible to both general and younger lay audience. Additionally, we employ specialized tokens and augmentation techniques to optimize the model's performance. Our methodology proved effective, earning us the 7th rank on the final leaderboard out of 57 participants.

1 Introduction

Lay summarization (i.e. summarization for non-expert audiences) helps make scientific literature understandable to non-experts. It simplifies complex technical information into clear, easy-to-understand language, promoting public understanding of research findings. The significance of lay summarization, which bridges the gap between scientific insights and public knowledge, has been increasingly recognized (Chandrasekaran et al., 2020; Goldsack et al., 2023).

To address the challenge of dense technical language in biomedical research papers, the BioLaySumm 2024 shared task (Goldsack et al., 2024) at the ACL 2024 BioNLP workshop focuses on turning biomedical research into lay summaries. These summaries need to be accurate and understandable to a broad audience, since they serve an important role in informing the public about scientific developments and avoiding the spread of misinformation. The shared task is based on data from two sources of biomedical articles: eLife and PLOS (Goldsack et al., 2022).

Our approach to this shared task is based on the BioBART-v2 model(Yuan et al., 2022), which has been demonstrated to be highly effective in summarizing biomedical content. On top of it, we employ a controllable generation technique using special tokens, in order to exploit in a single model the data from both eLife and PLOS and at the same time during inference align the produced summaries with the unique characteristics of the corresponding source, such as length, readability and level of abstraction.

In addition, towards improving the simplicity of the produced summaries, we employed augmentation to extend the eLife and PLOS data. We identified the most complex lay summaries in these datasets, and paired their source abstracts with summaries produced by GPT-4 (OpenAI et al.). To make it produce simple lay summaries, we used in-context learning providing to it examples of scientific articles targeted at children from the Science Journal for Kids.

2 The SJK Dataset

Science Journal for Kids (SJK) is a non-profit organization based in Texas that is dedicated to presenting scientific research in a manner that is accessible and appealing to children. They achieve this goal by digitally publishing on their web site[1], adaptations of scientific papers that are made to be kid-friendly. The adaptation process undertaken by SJK involves using common vocabulary and relatable examples, and then further validating and refining the adapted content for educational use. This process ensures that the content is not only accessible but also retains the educational value of the original scientific research.

The kid-friendly articles in the SJK web site are available in PDF format. To assemble the SJK dataset, we collected the PDFs of the articles and

[1] https://sciencejournalforkids.org/

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 793–803
August 16, 2024. ©2024 Association for Computational Linguistics

extracted their content. From this content we kept the abstracts and the links at the references section, pointing to the original scientific papers. We extracted DOI numbers from these links and attempted to retrieve the abstracts via the Semantic Scholar API. However, since the API often returned empty abstracts, we resorted also to an extensive scraping process on specific pages with a particular format to get the abstracts. This was not always feasible due to restrictions on scraping from certain sites, necessitating manual addition of links and abstracts to the dataset, highlighting the challenging nature of the data collection process.

We crawled the SJK web site on January 22, 2024. We initially collected all 306 articles from the SJK web site for potential future work, ensuring we had a comprehensive dataset to expand or refine as needed. From these, we eventually selected 285 articles based on their formatting suitability. Older versions had a completely different format that the scraping process couldn't recognize because it was based on the newer versions. Formatting issues included instances where the text from the abstract was cut off when scraped due to the two-column format or where scraping couldn't find the reference. We manually conducted additional checks to append missing text and locate references in these cases. Additionally, we prioritized articles that included references. Besides ensuring the credibility of the content, this allowed us to pair the kid-friendly articles with the corresponding scientific articles that inspired them.

Our final dataset[2] comprises 300 pairs, each consisting of an abstract from a scientific paper and its corresponding abstract from the children's article. We focused on abstracts because they provided comprehensive information suitable for our lay summarization task. For each article, we sourced the corresponding abstract from the first reference cited in the children's articles published by SJK, and in 25 cases, also the second reference, which are the original academic papers that the SJK articles are based on.

The articles were intentionally curated to encompass a wide array of subjects, specifically chosen to attract the scientific curiosity of young learners across disciplines such as biology, chemistry, and more. Table 1 illustrates the diversity of topics covered by both all SJK articles and our final dataset.

Table 1: Number of articles in the SJK web site and in our collection per category. Note that some articles belong to multiple categories.

Category	Ours	SJK
Biodiversity-And-Conservation	83	85
Health-And-Medicine	77	81
Biology	63	70
Energy-And-Climate	57	57
Social-Science	51	57
Water-Resources	48	48
Pollution	30	30
Food-And-Agriculture	25	26
Technology	20	23
Paleoscience	16	18
Chemistry	13	13
Physical Science	2	18

3 Our Approach

3.1 Model

Our approach employs the BioBART-v2 model. BioBART-v2 introduces significant improvements in its training methodology to advance its capabilities in the biomedical field. Unlike its precursor, which utilized a general-domain vocabulary, BioBART-v2 incorporates a specialized cross-domain vocabulary, substantially enlarging its lexicon to 85,401 tokens. This expansion is derived from Domain-Adaptive Pre-Training (DAPT) (Gururangan et al., 2020) on the PubMed abstracts corpus, resulting in a rich dataset that provides a more targeted pretraining foundation.

The construction of this vocabulary was achieved by merging the original BART's general-domain vocabulary with newly generated biomedical tokens, specifically designed from the PubMed corpus. This process yielded 60,000 additional tokens that, when combined with the existing vocabulary, boosted the model's capabilities for biomedical literature.

BioBART-v2, with its 400 million parameters, balances model complexity and computational feasibility, making it suitable for both research and practical applications. Fine-tuning it is straightforward due to its architecture, allowing targeted training on biomedical tasks with minimal computational resources. This adaptability makes it ideal for various applications in the biomedical domain, from information extraction to summarization.

[2] The dataset will be publicly available after notification of acceptance.

3.2 Data

We fine-tuned BioBART-v2 on the union of the two biomedical datasets offered by BioLaySumm 2024, i.e. PLOS and eLife. Preliminary experiments using a different model for each dataset led to inferior results. We used only the abstracts of the academic articles as sources. When properly written, the abstract of an article serves as a concise summary of the whole article, containing all the aspects needed for *translating* it into lay language. In addition, these abstracts align well with the content that was used for the pre-training of BioBART. Details about each of the two datasets follow.

The PLOS dataset comprises 26,291 articles from five peer-reviewed journals of the Public Library of Science (PLOS) publisher, covering diverse fields such as Biology, Computational Biology, Genetics, Pathogens, and Neglected Tropical Diseases. The lay summaries in this dataset are written by the authors of the articles themselves. These summaries typically range from 150 to 200 words in length. The dataset is divided into 24,773 training, 1,376 validation, and 142 testing articles.

The eLife dataset, contains 4,729 articles from the eLife biomedical journal, covering a wide array of topics in life sciences and medicine. In contrast to PLOS, eLife features lay summaries produced collaboratively by expert editors and the original authors. This collaboration resulted in summaries that are longer, more abstractive, and generally more readable. The dataset is divided into 4,346 documents for training, 241 for validation, and 142 for testing.

3.3 Data Augmentation

To improve the readability of the produced summaries, we extended the provided eLife and PLOS datasets by using GPT-4 to rewrite lay summaries of high complexity. To identify such summaries, we used three readability metrics: Flesch-Kincaid Grade Level (FKGL) (Flesch, 1948), Dale-Chall Readability Score (DCRS) (Chall and Dale, 1995), and Coleman-Liau Index (CLI) (Coleman and Liau, 1975). These metrics offer quantitative assessments of text complexity and measure the accessibility of the content across various age groups. Table 2 provides an interpretation of the FKGL metrics, illustrating how different score ranges correspond to reading and school levels. The DCRS and CLI scores similarly provide insights into the readability and complexity of the text. This approach is in alignment with the evaluation criteria of the BioLaySumm 2024 shared task.

Flesch-Kincaid Score	Reading Level
0 - 3	Kindergarten
3 - 6	Elementary
6 - 9	Middle School
9 - 12	High School
12 - 15	College
15 - 18	Post-grad

Table 2: Flesch-Kincaid Grade Level (FKGL) Metrics Interpretation

We specifically targeted the top 200 summaries from each of the eLife and PLOS datasets based on their highest FKGL scores, with the aim of simplifying them to reach the level of middle school students. In the eLife dataset, these summaries had average scores of FKGL 10.74, DCRS 12.39, and CLI 8.91, which correspond to high school and college reading levels. The PLOS dataset exhibited even higher complexity, with average scores of FKGL 14.73, DCRS 15.75, and CLI 10.86, aligning with college and post-graduate reading levels. These summaries, characterized by complex sentence structures and a high density of abstract ideas, were selected for augmentation to enhance their readability and accessibility for a middle school audience.

The DCRS and CLI metrics further support the interpretation of text complexity. DCRS scores above 10 indicate a higher level of text difficulty, often requiring college-level comprehension. Similarly, CLI scores, which reflect the number of characters per word and words per sentence, indicate higher complexity with scores above 8. The high DCRS and CLI scores of the selected summaries ensured that we focused on content that was particularly challenging, necessitating simplification for better accessibility.

To refine the summaries for children, we utilized the GPT-4 model via the OpenAI API, employing in-context learning via few-shot prompts to guide our augmentation pipeline. In particular, two randomly selected kids-friendly abstracts from the SJK dataset were used as examples during the augmentation process. These examples acted as guidelines, ensuring that the adapted summaries met the desired standards of simplicity. Additionally, the prompt asked to simplify the language and make it more accessible. An example of the prompt used for this purpose is illustrated below:

> *"You're explaining scientific concepts to a kid who's curious to learn. Keep all the important facts, but use easier words that are easier for kids to understand. Here are two examples of how to do it:"*
>
> *1. [A random kid-friendly abstract from the SJK dataset],*
> *2. [Another random kid-friendly abstract from the SJK dataset]*

Tables 3 and 4 present the mean scores of the original and augmented summaries. These scores demonstrate significant improvements in the readability of the augmented versions of the lay summaries.

Table 3: Mean readability scores for General (Targeted 200) and Kids summaries in the eLife dataset.

Category	FKGL	DCRS	CLI
Original	10.74	12.39	8.91
Augmented	7.90	8.99	7.33

Table 4: Mean readability scores for General (Targeted 200) and Kids summaries in the PLOS dataset.

Summary Type	FKGL	DCRS	CLI
Original	14.73	15.75	10.86
Augmented	8.57	9.07	7.50

3.4 Controllable Generation

Our methodology employs special tokens in the source abstracts to achieve two distinct controllable generation goals: i) adapt the produced summary towards the specific style of either of the two datasets, ii) guide the summary generation towards increased readability.

For the first goal, we use special tokens `<elife>` and `<plos>` to differentiate between the two datasets, as from the analysis in Sections 3.2 we know that expert-written eLife summaries are longer and more readable. For the second goal, we use special tokens `<general_lay_summary>` and `<kids_lay_summary>` to differentiate abstracts that are paired with original lay summaries from abstracts that are paired with augmented lay summaries adapted for children.

During training, we prepend each PLOS abstract with the `<plos>` tag and each eLife abstract with the `<elife>` tag. In addition, we prepend the augmented abstracts with the `<kids_lay_summary>` tag, while the rest of the abstracts are prepended with the `<general_lay_summary>` tag.

During inference, we again prepend each PLOS and eLife abstract with the corresponding `<plos>` and `<elife>` tags, while we experiment with including one or both of the `<general_lay_summary>` and `<kids_lay_summary>` tags to control the readability of the produced lay summary. Our final submission included both tags, as this led to the best results in the validation sets.

4 Results and Discussion

This section presents and discuss the results on the validation datasets provided by eLife and PLOS.

4.1 Experimental Setup

The fine-tuning process of BioBART-v2 was conducted using the Amazon Web Services (AWS) cloud platform. We utilized AWS S3 for storing model steps and output data. The fine-tuning tasks were executed on Amazon SageMaker, using a `p3.2xlarge` instance equipped with NVIDIA Tesla V100 GPU. More details on the experimental setup can be found in Appendix A.1.

We evaluated all the models using a combination of metrics to assess the *relevance*, *readability*, and *factuality* of the generated summaries, based on the BioLaySumm 2024 shared task. The relevance of the summaries was measured by metrics including ROUGE-1 (R-1), ROUGE-2 (R-2), ROUGE-L (R-L) (Lin, 2004), and BERTScore (Zhang* et al., 2019) to assess how well the content matched the original articles. Readability was evaluated through metrics such as the Flesch-Kincaid Grade Level, Dale-Chall Readability Score, Coleman-Liau Index, and LENS. Factuality was verified using AlignScore (Align S.) (Zha et al., 2023) and SummaC (Laban et al., 2021) to check the accuracy of the information presented in the summaries.

Our experimental results include the following variants:

- **Baseline:** This refers to the model's performance when trained using only the original scientific content of the eLife and PLOS datasets, without any additional data or special tokens.

Table 5: Experimental results on PLOS and eLife datasets.

Step	Approach	R-1	R-2	R-L	BertScore	FKGL	DCRS	CLI	LENS	Align S.	SummaC
					PLOS						
300	Baseline	0.488	0.171	0.449	0.850	14.149	11.142	14.370	73.706	0.778	0.636
300	Sp. Token	**0.494**	**0.173**	**0.454**	**0.865**	14.430	11.321	14.552	**74.978**	**0.790**	**0.647**
300	Sp. Token + Aug.	0.490	0.167	0.451	0.864	**13.839**	**10.914**	**13.336**	72.242	0.789	0.651
					eLife						
400	Baseline	0.479	0.133	0.453	0.838	10.979	8.813	11.541	72.445	0.622	0.539
400	Sp. Token	0.488	0.135	0.458	0.852	11.152	8.991	11.745	73.182	0.634	0.547
400	Sp. Token + Aug.	**0.491**	**0.135**	**0.462**	**0.851**	**10.636**	**8.750**	**11.284**	**73.707**	**0.640**	**0.548**
					Combined						
300+400	Sp. Token + Aug.	0.491	0.151	0.457	0.857	12.237	9.832	12.310	72.974	0.714	0.599

- **Sp. Token:** Represents the performance of the model when it has been added to with special tokens. This configuration does not include any augmented data.

- **Sp. Token + Augmented:** This configuration includes the use of special tokens, as mentioned above, along with the data augmentation strategy.

4.2 Results

This subsection highlights the summaries produced by our models at their best-performing steps during the competition. These results demonstrate the effectiveness of our specialized configurations, including the use of special tokens and augmented data, aimed at improving both the accessibility and accuracy of the summaries.

We detail the performance metrics for the PLOS and eLife, illustrating significant improvements in readability as a result of our modeling efforts, as shown in Table 5. Two different checkpoints were selected for the final summaries of the eLife and PLOS to optimize generation in line with the unique characteristics and challenges of each dataset. The chosen checkpoints reflect points where the model achieved an optimal balance between relevance, readability, and factual accuracy specific to each dataset. A more detailed analysis regarding each of the relevance, readability and factuality metrics along with detailed plots illustrating the different training steps can be found inA.2.

The use of special tokens consistently improved relevance scores across both datasets, indicating their effectiveness in helping the model understand the context and semantics better. Without special tokens, the model's relevance scores were notably lower, showing that it struggled to capture the essential details of the scientific content. This pattern was observed in both the eLife and PLOS datasets, highlighting the critical role of special tokens in enhancing the model's performance.

5 Conclusion

Our approach to the BioLaySumm 2024 shared task showcases BioBART's ability to simplify complex biomedical research articles into accessible lay summaries. By fine-tuning BioBART with specialized tokens and data augmentation techniques, we generated readable summaries for specific audiences, including younger readers.

A key aspect of our methodology was the use of specialized tokens to precisely control the characteristics of each dataset and audience. Additionally, we enriched our dataset with kid-friendly content from the Science Journal for Kids, enabling us to produce summaries that effectively bridge the gap between scientific complexity and public understanding. Our experimental results highlight the effectiveness of our approach, especially in improving the readability and relevance of the summaries.

While our methodology significantly improved readability and relevance, maintaining factual accuracy remains a challenge. Ensuring the factuality of lay summaries is especially critical in the biomedical field, where accuracy is important.

Our model achieved an 7th place out of 55 participants, demonstrating its validity in managing diverse and complex summarization tasks. This achievement shows the potential of our techniques in making scientific knowledge more accessible to the general public and children.

6 Limitations

In this work, we employed the BioBART model with specialized tokens and data augmentation techniques to generate lay summaries of biomedical research articles. While our approach improved the readability and relevance of the summaries, we did not explicitly analyze the factual accuracy of the generated summaries, which remains a critical issue in the biomedical domain. The introduction of augmented data, while beneficial for readability, sometimes compromised content relevance and factual accuracy. To improve the quality of our training examples, future research could integrate factuality metrics to evaluate the accuracy of generated summaries and use post-editing techniques or human review to remove inaccurate content.

Acknowledgements

AWS resources were provided by the National Infrastructures for Research and Technology GRNET and funded by the EU Recovery and Resiliency Facility.

References

Jeanne Sternlicht Chall and Edgar Dale. 1995. *Readability Revisited: The New Dale-Chall Readability Formula*. Brookline Books. Google-Books-ID: 2nbuAAAAMAAJ.

Muthu Kumar Chandrasekaran, Guy Feigenblat, Eduard Hovy, Abhilasha Ravichander, Michal Shmueli-Scheuer, and Anita de Waard. 2020. Overview and insights from the shared tasks at scholarly document processing 2020: CL-SciSumm, LaySumm and LongSumm. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 214–224, Online. Association for Computational Linguistics.

Meri Coleman and T. L. Liau. 1975. A computer readability formula designed for machine scoring. 60(2):283–284. Place: US Publisher: American Psychological Association.

R. Flesch. 1948. A new readability yardstick. 32(3):221–233.

Tomas Goldsack, Zheheng Luo, Qianqian Xie, Carolina Scarton, Matthew Shardlow, Sophia Ananiadou, and Chenghua Lin. 2023. Overview of the biolaysumm 2023 shared task on lay summarization of biomedical research articles.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360. Association for Computational Linguistics.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2021. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Preprint*, arxiv:2111.09525 [cs].

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, C. J. Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. GPT-4 technical report. *Preprint*, arxiv:2303.08774 [cs].

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Hongyi Yuan, Zheng Yuan, Ruyi Gan, Jiaxing Zhang, Yutao Xie, and Sheng Yu. 2022. BioBART: Pretraining and evaluation of a biomedical generative language model. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 97–109, Dublin, Ireland. Association for Computational Linguistics.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. *Preprint*, arxiv:2305.16739 [cs].

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2019. BERTScore: Evaluating text generation with BERT.

A Appendix

A.1 Experimental Setup

Here, we present additional details regarding the experimental setup.

A.1.1 Distribution of text lengths

Firstly, as part of our configuration, we determined that the maximum input length would be set at 400 words based on the distribution of text lengths across our datasets, as shown in Table 6. This table provides the 95th percentile of text lengths and the percentage of texts that are 400 words or fewer, demonstrating that the chosen maximum input length effectively covers the majority of the data.

Table 6: Distribution of text lengths in the validation set.

Dataset	95th Perc. Length (words)	% $\leq$ 400 words
eLife Abstracts	186.09	100.0
PLOS Abstracts	368.00	97.02

A.1.2 Training Configuration

We fine-tuned and configured parameters using the Hugging Face Transformers library (Wolf et al., 2020) to ensure maximum efficiency. After a limited preliminary exploration of hyperparameter values on the validation sets of eLife and PLOS, we established the most effective settings. We set the learning rate at 1×10^{-5} to balance the speed and stability of the learning process.

We chose a batch size of 4 for both training and evaluation to optimize GPU memory usage. The model underwent training over 15 epochs, with evaluations and model savings every 50 steps to consistently monitor and evaluate progress.

A key component was the use of gradient accumulation, where we applied 64 steps. This method effectively increases the batch size to 256 (4 times 64), allowing us to handle larger batches and stabilize the training dynamics without requiring additional memory.

Thus, the number of data samples processed at each checkpoint can be determined by the following formula:

$$\begin{aligned}\text{Train Batch Size} \times \text{Gradient Acc. Steps} \times \text{Save Steps} \\ = 4 \times 64 \times 50 \\ = 12,800\end{aligned}$$

A.2 Detailed Analysis and Training Plots

Here, we provide a more detailed analysis regarding the effectiveness of each approach across different steps in terms of relevance, readability, and factuality. For the sake of presentation clarity, we selected three indicative training checkpoints for detailed examination, which summarize the whole training process. We used different numbers of steps for the eLife and PLOS datasets to better present the key outcomes for each dataset.

A.2.1 Relevance

The relevance of the generated summaries is measured using ROUGE scores. As shown in Tables 7 and 8, the relevance for the eLife dataset significantly improved with training, reflecting the increasing ROUGE scores. This improvement suggests that the eLife dataset, which includes longer, and more readable lay summaries written by expert editors, provides new and varied content that the model effectively learns from during training.

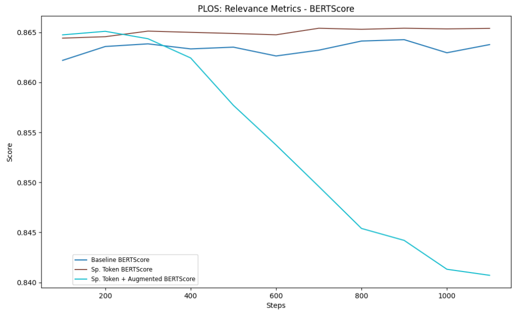


Figure 1: BERTScore relevance metric for PLOS articles.

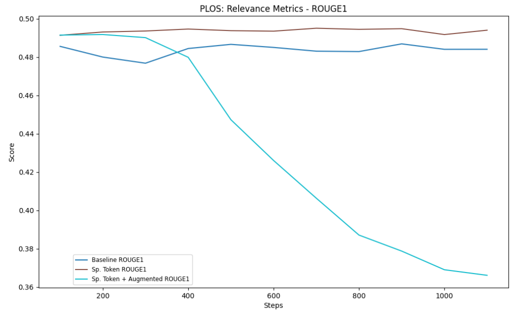


Figure 2: ROUGE-1 relevance metric for PLOS articles.

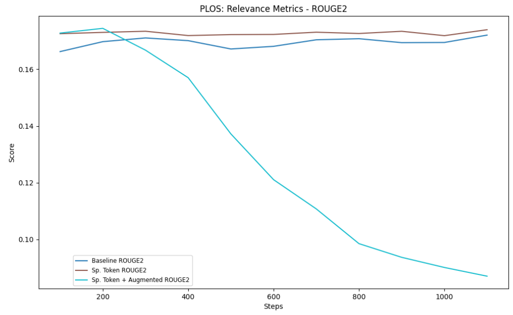


Figure 3: ROUGE-2 relevance metric for PLOS articles.

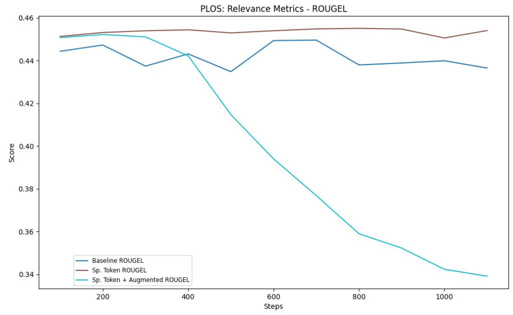


Figure 4: ROUGE-L relevance metric for PLOS articles.

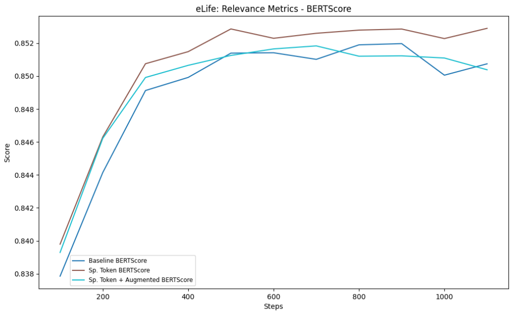


Figure 5: BERTScore relevance metric for eLife articles.

Table 7: Metrics for the eLife dataset at selected steps

Step	Approach	R-1	R-2	R-L	BERT	FKGL	DCRS	CLI	LENS	Align	SummaC
100	Baseline	0.376	0.090	0.338	0.838	13.29	10.44	14.39	56.70	0.760	0.625
100	Sp. Token	0.380	0.093	0.351	0.840	13.58	10.65	14.82	58.66	0.761	0.646
100	Sp. Token + Aug.	0.373	0.088	0.345	0.839	13.52	10.66	14.91	59.08	0.766	0.649
500	Baseline	0.488	0.134	0.451	0.850	10.63	8.75	11.42	71.53	0.641	0.543
500	Sp. Token	0.493	0.138	0.463	0.853	10.96	8.95	11.55	73.96	0.642	0.556
500	Sp. Token + Aug.	0.494	0.136	0.465	0.851	10.15	8.56	10.69	76.15	0.608	0.540
900	Baseline	0.493	0.137	0.453	0.851	10.55	8.72	10.89	72.58	0.619	0.519
900	Sp. Token	0.501	0.141	0.471	0.853	10.75	8.82	11.30	74.91	0.620	0.527
900	Sp. Token + Aug.	0.497	0.138	0.468	0.851	9.64	8.24	10.02	78.22	0.583	0.546

Table 8: Metrics for the PLOS dataset at selected steps

Step	Approach	R-1	R-2	R-L	BERT	FKGL	DCRS	CLI	LENS	Align	SummaC
100	Baseline	0.476	0.167	0.437	0.864	14.03	10.96	14.17	72.12	0.782	0.632
100	Sp. Token	0.491	0.173	0.451	0.864	14.51	11.32	14.41	74.89	0.784	0.643
100	Sp. Token + Aug.	0.491	0.173	0.451	0.865	14.50	11.29	14.32	74.95	0.783	0.641
300	Baseline	0.480	0.169	0.447	0.863	14.27	10.99	14.28	73.48	0.788	0.632
300	Sp. Token	0.494	0.173	0.454	0.865	14.43	11.32	14.55	74.98	0.790	0.647
300	Sp. Token + Aug.	0.490	0.167	0.451	0.864	13.84	10.91	13.34	72.24	0.789	0.651
600	Baseline	0.479	0.170	0.438	0.863	14.21	10.94	14.26	72.78	0.794	0.637
600	Sp. Token	0.493	0.172	0.454	0.865	14.61	11.37	14.73	75.10	0.796	0.654
600	Sp. Token + Aug.	0.426	0.121	0.394	0.854	11.36	9.16	10.44	58.34	0.791	0.657

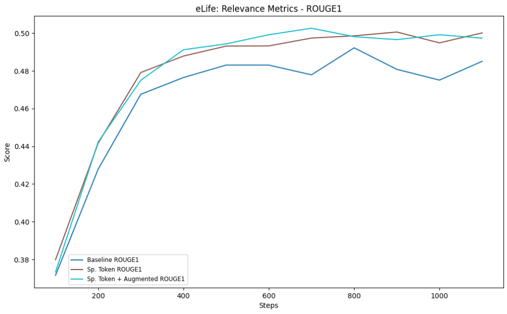


Figure 6: ROUGE-1 relevance metric for eLife articles.

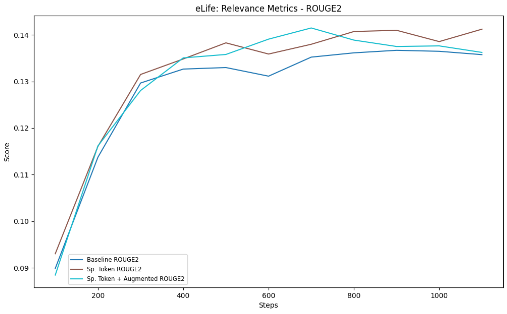


Figure 7: ROUGE-2 relevance metric for eLife articles.

For the PLOS dataset, however, the relevance did not show significant improvement with training, suggesting that the model might have already been exposed to similar data during its initial training on the PubMed archive, where PLOS articles are included (for example here: PubMed archive).

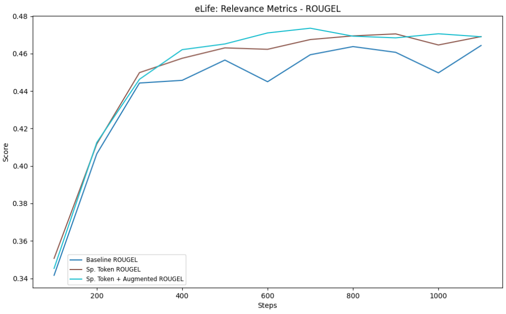


Figure 8: ROUGE-L relevance metric for eLife articles.

Additionally, the introduction of augmented data led to a decline in relevance at later steps, suggesting that the diversity brought by augmentation may complicate content relevance when the model has already encountered similar datasets.

A.2.2 Readability

Readability generally improved across successive training steps, as indicated by the FKGL, CLI, DCRS, and LENS scores in Tables 7 and 8. For the eLife dataset, the use of special tokens, along with training on new, unseen data, helped reduce complexity, making the summaries easier to understand. This consistent improvement is likely due to the nature of eLife's longer, more detailed, and editor-written summaries. Special tokens, and also augmented data, further aided this process by helping the model capture and organize the relevant contextual information more effectively.

For the PLOS dataset, while training did not significantly affect relevance or factuality, it did improve readability. This indicates that even if the model had seen similar data before, the fine-tuning process still contributed to producing more readable summaries. Augmented data helped improve readability scores, simplifying the text.

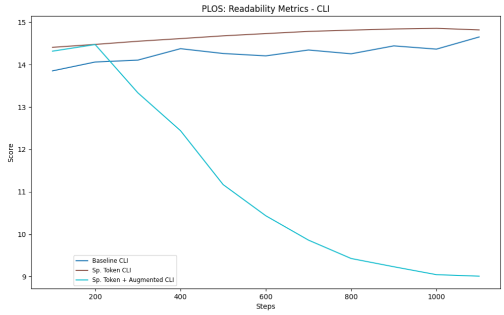


Figure 9: Coleman-Liau Index readability metric for PLOS articles.

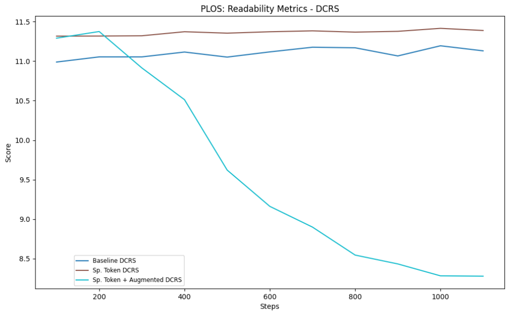


Figure 10: DCRS readability metric for PLOS articles.

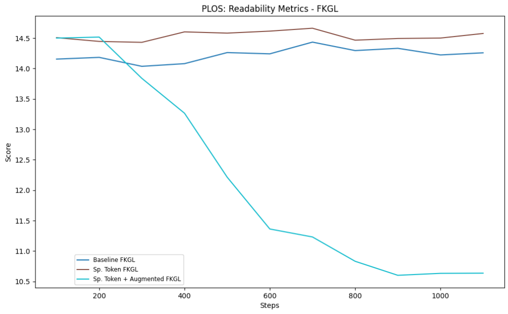


Figure 11: Flesch-Kincaid Grade Level (FKGL) readability metric for PLOS articles.

A.2.3 Factuality Metrics

Factuality metrics reveal a complex pattern of performance. For the eLife dataset, while factuality

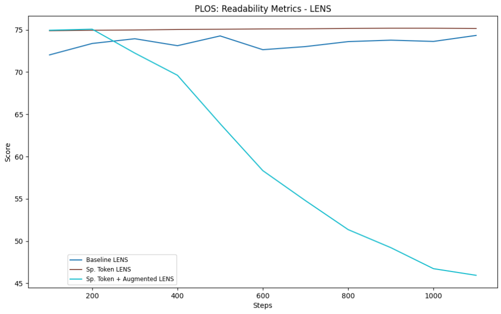


Figure 12: LENS readability metric for PLOS articles.

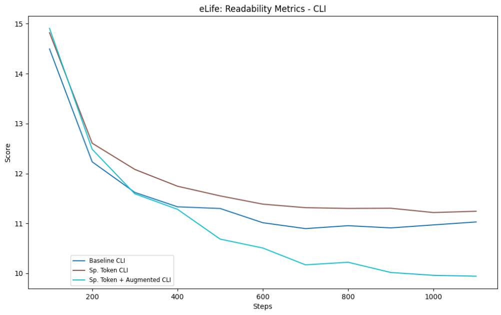


Figure 13: Coleman-Liau Index readability metric for eLife articles.

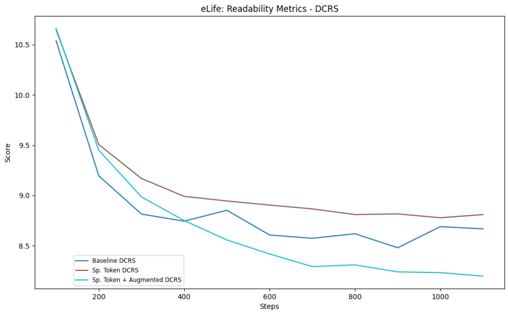


Figure 14: DCRS readability metric for eLife articles.

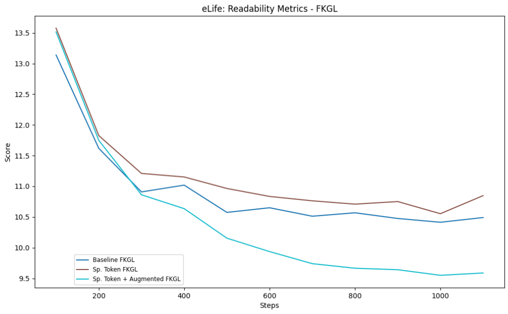


Figure 15: Flesch-Kincaid Grade Level (FKGL) readability metric for eLife articles.

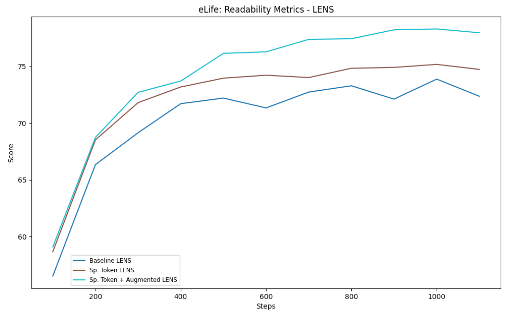


Figure 16: LENS readability metric for eLife articles.

scores showed some improvement with training, the introduction of augmented data sometimes led to a decline in factuality, especially in later steps. This suggests challenges in maintaining accuracy when introducing more diverse training data, particularly for a dataset that is initially more abstractive.

For the PLOS dataset, factuality scores did not consistently improve with training and decreased in later steps, particularly with the introduction of augmented data. This suggests that adding more diverse data did not help maintain factual accuracy and may have introduced complexity.

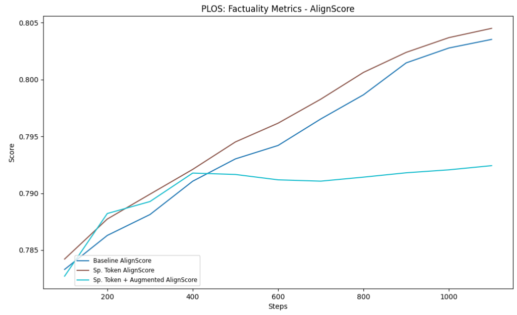


Figure 17: Alignment Score factuality metric for PLOS articles.

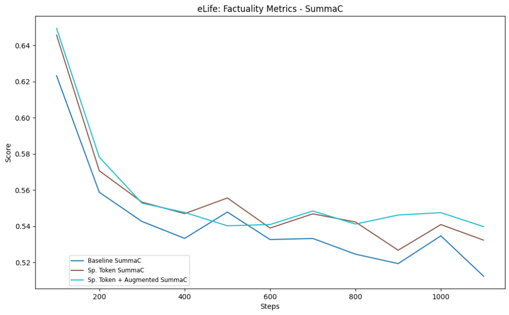


Figure 18: SummaC factuality metric for eLife articles.

SINAI at BioLaySumm: Self-Play Fine-Tuning of Large Language Models for Biomedical Lay Summarisation

Mariia Chizhikova, Manuel C. Díaz Galiano
L. Alfonso Ureña López and **M. Teresa Martín Valdivia**
Department of Computer Science, University of Jaén
Campus Las Lagunillas, s/n, 23071, Jaén, Spain
mchizhik@ujaen.es

Abstract

An effective disclosure of scientific knowledge and advancements to the general public is often hindered by the complexity of the technical language used in research which often results very difficult, if not impossible, for non-experts to understand. In this paper we present the approach developed by the SINAI team as the result of our participation in BioLaySumm shared task hosted by the BioNLP workshop at ACL 2024. Our approach stems from the experimentation we performed in order to test the ability of state-of-the-art pre-trained large language models, namely GPT 3.5, GPT 4 and Llama-3, to tackle this task in a few-shot manner. In order to improve this baseline, we opted for fine-tuning Llama-3 by applying parameter-efficient methodologies. The best performing system which resulted from applying self-play fine tuning method which allows the model to improve while learning to distinguish between its own generations from the previous step from the gold standard summaries. This approach achieved 0.4205 ROUGE-1 score and 0.8583 BERTScore.

1 Introduction

Science outreach and scientific advocacy are crucial for the development of the science itself, as most funding still comes from public sources and thus demands public's support. Furthermore, science is central to most of the grand challenges of today's society, such as climate change, economic productivity, health and new drug discovery. These factors highlight the relevance of making information about scientific advancements accessible for general public. Moreover, it also may help the public make sound and informed choices about issues like participating in a clinical trial or getting a vaccination (Varner, 2014).

Nevertheless, even with an increased online availability of scientific publications, accessing the information from these sources remains a challenging task for non-experts due to the technical language and specific terminology used to write scientific work. One viable solution for addressing the informational requirements of the general public or gatekeepers like journalists are plain language summaries or lay summaries - a format that presents scientific research in an easily understandable manner for non-experts (King et al., 2017). However, manual generation of lay summaries is a tedious and costly process that involves contracting expert writers specialised in science outreach. For this reason, the development of effective automatic lay summarisation systems is attracting an increasing amount of attention of the Natural Language Processing (NLP) researchers (Ermakova et al., 2022).

BioLaySumm shared task held on the BioNLP 2024 workshop at ACL brings the community effort to tackle the task of automatic abstractive summarisation of biomedical articles for non-technical audiences by leveraging the extensive work done by the creators of eLife (King et al., 2017) and the Public Library of Science (PLOS) database in manually composing lay summaries.

This paper presents the methodology developed by the SINAI team as a part of our participation in the BioLaySumm shared task. Our experimentation involved comparison between few-shot learning (FSL) of instruction-tuned pre-trained large language models (LLMs), parameter-efficient tuning of open-source pre-trained LLMs and Self-Play fine tuning (SPIN) methodology which allows the model to improve while learning to distinguish between its own generations from the previous step form the gold standard summaries. The latter mentioned approach resulted to be the highest-scoring submission among all made by our team achieving 0.4205 ROUGE-1 score and 0.8583 BERTScore.

The remainder of the paper is structured as follows: Section 2 provides a concise description of the data utilized for this task. Section 3 details

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 804–809
August 16, 2024. ©2024 Association for Computational Linguistics

the systems developed by our team for the official evaluation. The details of the evaluation process and the results are presented in Section 4. Finally, Section 5 concludes our work.

2 Data

The organisers of the BioLaySumm shared task put at the disposal of the participants two datasets, PLOS and eLife, each of which consisted of biomedical research articles (including their technical abstracts) and their expert-written lay summaries (Goldsack et al., 2022). As for the dimensions of the data, while PLOS may be considered large-scale dataset with 24,773 instances for training and 1,376 for validation, eLife is a medium-scale dataset comprised of 4,246 training instances and 241 validation instances.

One important difference between the two datasets lies in the process of its generation. The Public Library of Science (PLOS) is a publisher that hosts peer-reviewed journals several of which require authors to submit an 150-200 word long *author summary* alongside their work. In contrast with this, eLife is an open-access journal that started creating plain-language summaries of its research articles in 2021 (King et al., 2017). As a result, lay summaries from two datasets differ from each other according to several characteristics, such as length and the extent to which they are abstractive. As can be seen on Table 1, which presents the statistics of token counts[1], eLife lay summaries and abstracts are almost twice as long. Furthermore, the authors claim that lay summaries of eLife appear to be significantly more abstractive based on the fact that these consistently contain more novel *n*-grams than abstracts across both datasets (King et al., 2017).

3 Methods

This section details the implementation of systems presented by our team for the official evaluation. We tested three approaches to lay summary generation: FSL of instruction-tuned models, parameter efficient fine-tuning of text generation pre-trained models and a novel method of fine-tuning of LLMs called SPIN.

3.1 Few-shot learning

FSL of pre-trained LLMs like GPT-3.5 proved to be a robust approach to the task of lay summarisation during the previous editions of the BioLaySumm shared task (Turbitt et al., 2023). For this reasons we decided to perform experiments with both closed, such as GPT-3.5 (Brown et al., 2020) and GPT-4 (Achiam et al., 2023) and open models, namely Llama-3 (AI@Meta, 2024). The GPT models were accessed via OpenAI API[2], while the Llama-3-8B model was run on 1 NVIDIA Tesla V100-PCIE-32GB by making use of `transformers` Python library (Wolf et al., 2019).

Our team adopted 2-shot-prompting approach that introduced one randomly selected example from each of the datasets and injected a mention of the dataset to which each text abstract belonged. Thus, our method creates one system for the two datasets. The details regarding the prompt can be found in Appendix A.

While performing the experiments, we empirically found out that outputs from Llama-3 occasionally contained definition of what a lay summary is, repetition of prompt's content and `LAY SUMMARY` prefix. In order to remove this noise from the generation, we designed a post-processing procedure based on regular expressions.

3.2 Fine-tuning

While FSL allows adaptation of the model to a task without the need of further training of an LLM, fine-tuning involves training the model using additional, task-specific data.

We selected Llama-3-8B[3], the newest open LLM at the time of system's development process, for fine-tuning employing the QLoRA method (Dettmers et al., 2023), which minimizes memory usage and the number of trainable parameters by backpropagating gradients through a frozen, 4-bit quantized pre-trained LLM into Low Rank Adapters (LORA).

In order to obtain a single model capable of generating lay summaries for instances of both datasets, we trained Llama-3-8B on the data obtained by merging both training dataset into one.

As for the hyperparameters set for training, the LoRA alpha was set to 16, LoRA dropout was

[1] We used the tokenizer of BioMistral-7B model to split the texts into tokens

[2] `https://openai.com/api/`

[3] `https://huggingface.co/meta-llama/Meta-Llama-3-8B`

		Lay summaries			Abstracts		
Dataset	Subset	Avg(STD)	Min	Max	Avg(STD)	Min	Max
eLife	Train	479.53 (84.69)	226	875	255.87 (45.67)	95	798
	Val	486.97 (93.62)	285	894	255.55 (43.24)	120	524
PLOS	Train	269.11 (58.15)	18	675	400.85 (111.11)	18	1198
	Val	269.53 (57.46)	72	530	404.1 (109.78)	112	877

Table 1: Token count statistics across two datasets.

equal to 0.1, LoRa rank - to 64 and the batch size was set to 1. The number of training epochs was initially set to 10, but an early stopping mechanism was implemented to prevent overfitting by stopping the training when validation loss does not decrease for 3 consecutive epochs. This allowed us to determine that one epoch was optimal for this kind of training.

3.3 Self-play fine-tuning

A significant advancement in LLMs performance is often achieved by applying post-pretraining alignment with mode desirable behaviour by using such techniques as Direct Preference Optimization (DPO) (Rafailov et al., 2024). Nevertheless, most alignment methods require a large volume of high-quality human-annotated data, which was not available for this challenge. For this reason, we opted for experimenting with SPIN (Chen et al., 2024), a novel fine-tuning method which begins from a supervised fine-tuning model, Llama-3-8B-instruct [4] denoted by $\mathbf{p}_{\theta_t}$ which is employed to generate responses $\mathbf{y}'$ to the prompt $\mathbf{x}$ in the gold standard dataset, $\mathbf{y}$. The objective is to find a new LLM $\mathbf{p}_{\theta_t+1}$ capable of distinguishing $\mathbf{y}'$ from $\mathbf{y}$.

We employed this method to train a QLoRA adapter in order to be able to perform training on a single NVIDIA Ampere A100 50Gb GPU. We applied this method to fine-tune the model on each dataset separately, which resulted in two different adapters for eLife and PLOS datasets respectively.

4 Evaluation

This section presents the results of the official evaluation campaign that was carried out by the organizers by assessing the predictions made by our system on 142 articles for each of the two datasets.

4.1 Evaluation metrics

The generated summaries were evaluated across three aspects: Relevance, Readability and Factuality. To assess the relevance, n-gram based metrics (ROUGE 1, 2 and L) and semantic similarity metrics were calculated (BERTScore). In order to evaluate the readability Flesch-Kincaid Grade Level (FKGL) and Dale-Chall Readability Score (DCRS), Coleman-Liau Index (CLI), and LENS were used. Finally, to assess the factuality, the organisers calculated AlignScore and SummaC (Goldsack et al., 2024).

4.2 Results

Table 2 presents the details of the relevance metrics scored by each of the presented systems. Among the FSL experiments, Llama-8B-Instruct demonstrated the lowest performance among the models evaluated. Nonetheless given the unknown number of parameters of GPT-3.5-turbo and GPT-4-turbo, as well as the lack of information about whether these two closed-source systems add any post-processing to their outputs, it is difficult to draw conclusions about the performance of the models themselves. Nevertheless, we could empirically observe that the generations of both GPT-3.5 and GPT-4 are always complete and concluded texts, while Llama-3-8B-Instruct often outputted truncated or, on the contrary, noisy at the end of the sequence text. For this reason, as we noted previously, we introduced a rule-based post-processing procedure, which resulted in achieving the highest relevance scores for the eLife dataset.

As for GPT-3.5 and GPT-4, we were not able to find a substantial difference in overall performance between those two systems in the FSL setting. However, it is noticeable that GPT-3.5 showcased one of the best performances in terms of BERTScore for the eLife dataset and outperformed GPT-4 in all relevance metrics for PLOS dataset.

Comparing the results from the two fine-tuning methods employed, we can see that SPIN

[4] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

fine-tuning of Llama3-8B-Instruct outperformed QLoRA in generating lay summaries for PLOS dataset. This can result from a larger amount of data available for this dataset, which makes SPIN to produce a more robust model. Nevertheless, for a much smaller dataset such as eLife, training a separate adapter with SPIN did not yield a performance improvement, while merging eLife with PLOS for training a universal QLoRa adapter for Llama-3-8B resulted to be a better solution in terms of relevance metrics.

As for readability and factuality, Table 3 presents the values these metrics. Overall, most of the systems produced less complex text that the reference lay summaries for PLOS datasets were reported to be (14.76, 10.91 and 15.90 for FKGL, DCRS and CLI, respectively) (Goldsack et al., 2022), while the reported lack of complexity for eLife's lay summaries (10.92, 8.83 and 12.51 for FKGL, DCRS and CLI, respectively) was more difficult to achieve even with our best system in terms of the readability metrics, namely FSL with Llama-3-8B-Instruct and rule-based post-processing.

Notably, the GPT-4 model, among all the presented systems, was the one that produced generally more complex text than others, while scoring one of the lowest values for factuality metrics. With regard to that, there can be perceived a trade-off between factuality and readability, with higher ranked models in terms of readability criteria achieving lesser factuality scores and vice-versa. The best performing model in terms of factuality resulted from fine-tuning of Llama-3-8B with QLoRA.

5 Conclusions

In this study, we explored various methodologies to generate lay summaries of biomedical articles, an important task for improving public accessibility to scientific information. Our participation in the BioLaySumm shared task at the BioNLP2024 workshop involved experimenting with FSL, parameter-efficient tuning and SPIN methods. Among these, SPIN fine-tuning demonstrated the highest performance in terms of relevance metrics, achieving a 0.4205 ROUGE-1 score and 0.8583 BERTScore.

The evaluation of readability and factuality revealed a trade-off between these two aspects. Models that generated more readable texts tended to have lower factuality scores, with the GPT-4 based FSL systems exemplifying this trend. Conversely, the fine-tuned Llama-3-8B with QLoRA achieved the best factuality scores while getting fairly good readability scores as well, indicating its potential for producing accurate and readable summaries.

6 Limitations

The limitations of the presented approaches stem from the inherent characteristics and potential biases of the pre-trained models they are based on. Specifically, models like Llama-3-8B-Instruct, GPT-3.5, and GPT-4 were pre-trained on extensive text datasets, which were not thoroughly evaluated for existing biases. Consequently, these models may generate inappropriate content or replicate biases present in the underlying data. Therefore, it is crucial to conduct comprehensive evaluations of safety and fairness concerns before deploying these systems in any practical applications.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AI@Meta. 2024. Llama 3 model card.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. 2024. Self-play fine-tuning converts weak language models to strong language models. *arXiv preprint arXiv:2401.01335*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. In *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115. Curran Associates, Inc.

Liana Ermakova, Eric SanJuan, Jaap Kamps, Stéphane Huet, Irina Ovchinnikova, Diana Nurbakova, Sílvia Araújo, Radia Hannachi, Elise Mathurin, and Patrice Bellot. 2022. Overview of the clef 2022 simpletext lab: Automatic simplification of scientific texts. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 470–494. Springer.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing*

System	Dataset	ROUGE-1↑	ROUGE-2↑	ROUGE-L↑	BERTScore↑
Llama-3 + QLoRa on merged datasets	PLOS	0.4038	0.1419	0.3766	0.8495
	eLife	0.418	0.1007	0.3969	0.836
	Overall	0.4109	0.1213	**0.3867**	0.8428
Chat-GPT-3.5-turbo FSL	PLOS	0.4216	0.1076	0.3805	0.8603
	eLife	0.3719	0.0987	0.3418	0.8491
	Overall	0.3969	0.1032	0.3612	0.8547
Chat-GPT-4-turbo FSL	PLOS	0.4016	0.0834	0.3637	0.8548
	eLife	0.4139	0.0941	0.3771	0.849
	Overall	0.4077	0.0888	0.3704	0.8519
Llama-3-8B-Instruct FSL	PLOS	0.2958	0.067	0.2757	0.8045
	eLife	0.4118	0.0933	0.3892	0.8118
	Overall	0.3537	0.0802	0.3324	0.8082
Llama-3-8B-Instruct FSL + post-processing	PLOS	0.3904	0.0896	0.3609	0.8536
	eLife	**0.4262**	**0.1091**	**0.3997**	**0.8493**
	Overall	0.4083	0.0994	0.3803	0.8514
Llama-3-8B SPIN	PLOS	**0.4591**	**0.1485**	**0.418**	**0.8692**
	eLife	0.3819	0.1013	0.3527	0.8474
	Overall	**0.4205**	**0.1249**	0.3853	**0.8583**

Table 2: Detailed scores of the relevance metrics' values obtained by the systems presented by the SINAI team

System	Dataset	FKGL↓	DCRS↓	CLI↓	LENS↑	AlignScore↑	SummaC↑
Llama-3 + QLoRa on merged datasets	PLOS	12.5437	9.2242	14.5609	53.2788	**0.7663**	**0.7752**
	eLife	11.7704	8.5377	13.5612	58.2338	**0.747**	**0.7216**
	Overall	12.157	8.881	14.0611	55.7563	**0.7567**	**0.7484**
Chat-GPT-3.5-turbo FSL	PLOS	12.5183	10.015	13.9546	78.4641	0.7311	0.5396
	eLife	12.9662	9.8737	14.2676	77.5932	0.737	0.531
	Overall	12.7423	9.9441	14.111	78.0286	0.734	0.5353
Chat-GPT-4-turbo FSL	PLOS	14.4599	11.0353	15.8399	72.3903	0.6255	0.4635
	eLife	14.6803	10.923	15.893	71.65	0.6598	0.4692
	Overall	14.5701	10.9791	15.8664	72.0301	0.6427	0.4663
Llama-3-8B-Instruct FSL	PLOS	12.2824	**8.2744**	12.0751	46.8115	0.4857	0.4943
	eLife	12.3472	**8.3412**	12.8586	50.1217	0.5071	0.5057
	Overall	12.315	**8.3078**	12.4668	48.4662	0.4964	0.4999
Llama-3-8B-Instruct FSL + post-processing	PLOS	**10.8711**	8.6886	**12.0567**	**81.3139**	0.6274	0.5167
	eLife	**11.0275**	8.5286	**12.4404**	**81.1903**	0.6603	0.5341
	Overall	**10.9493**	8.6086	**12.2486**	**81.2521**	0.6439	0.5254
Llama3 SPIN	PLOS	12.8408	10.667	14.8027	73.3912	0.7521	0.5505
	eLife	11.6155	9.0522	12.8268	80.5002	0.6713	0.5291
	Overall	12.2281	9.8609	13.8148	76.9457	0.7117	0.5398

Table 3: Detailed scores of the readability and factuality metrics' values obtained by the systems presented by the SINAI team

and BioNLP Shared Tasks, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Stuart RF King, Emma Pewsey, and Sarah Shailes. 2017. Plain-language summaries of research: An inside guide to elife digests. *eLife*, 6:e25410.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Oisín Turbitt, Robert Bevan, and Mouhamad Aboshokor. 2023. Mdc at biolaysumm task 1: Evaluating gpt models for biomedical lay summarization. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 611–619.

Johanna Varner. 2014. Scientific Outreach: Toward Effective Public Engagement with Biological Science. *BioScience*, 64(4):333–340.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

A Prompt engineering

This appendix section details the prompts used for each of the lay summary generation methods described in this paper.

A.1 Few-shot learning

For the FSL approach we randomly selected 2 examples from the training sets of each database. The example A.1 presents the prompt template used for FSL generation.

Example A.1. You are an expert in generating of lay summaries - more readable summaries of scientific papers that are accessible to the general public. You will be given abstracts of scientific paper either from PLOS of eLife databases and will return only lay summaries like in the following examples:

This document is from {source} database, create the lay summary from abstract.

ABSTRACT: {Example abstract 1}

LAY SUMMARY: {Example lay summary 1}

This document is from {source} database, create the lay summary from abstract.

ABSTRACT: {Example abstract 2}

LAY SUMMARY: {Example lay summary 2}

This document is from {source} database, create the lay summary from abstract.

ABSTRACT: {Test set abstract}

LAY SUMMARY:

RAG-RLRC-LaySum at BioLaySumm: Integrating Retrieval-Augmented Generation and Readability Control for Layman Summarization of Biomedical Texts

Yuelyu Ji[1], **Zhuochun Li**[1], **Rui Meng**[2], **Sonish Sivarajkumar**[1],
Yanshan Wang[1], **Zeshui Yu**[1], **Hui Ji**[1], **Yushui Han**[1],
Hanyu Zeng [1], **Daqing He**[1]

[1]University of Pittsburgh, [2]Salesforce Research

Abstract

This paper introduces the RAG-RLRC-LaySum framework, designed to make complex biomedical research understandable to laymen through advanced Natural Language Processing (NLP) techniques. Our Retrieval Augmented Generation (RAG) solution, enhanced by a reranking method, utilizes multiple knowledge sources to ensure the precision and pertinence of lay summaries. Additionally, our Reinforcement Learning for Readability Control (RLRC) strategy improves readability, making scientific content comprehensible to non-specialists. Evaluations using the publicly accessible PLOS and eLife datasets show that our methods surpass Plain Gemini model, demonstrating a 20% increase in readability scores, a 15% improvement in ROUGE-2 relevance scores, and a 10% enhancement in factual accuracy. The RAG-RLRC-LaySum framework effectively democratizes scientific knowledge, enhancing public engagement with biomedical discoveries [1].

1 Introduction

Biomedical research encompasses crucial discoveries, ranging from everyday health concerns to significant disease outbreaks. Such studies are essential not only for scientists and doctors but also for journalists and the general public. However, the specialized and complex language typical in these studies often renders the content incomprehensible to those without a scientific background Thoppilan et al. (2022). To address this issue, the development of automated lay summaries have become increasingly important Goldsack et al. (2023b,a). This initiative aims to summarize the detailed aspects of biomedical research into summaries that are both comprehensible and devoid of complicated jargon. Although these systems show great potential, doubts about their accuracy are a major obstacle to their widespread useGabriel et al. (2020); Maynez et al. (2020); Yang et al. (2024); Li et al. (2023b, 2024); Wang et al. (2024). Our framework integrates specific external explanations for complex terms to further enhance content simplification. In response to the concerns about the integrity of summarized information, our framework employs a “knowledge retrieval” approach within the Retrieval-Augmented Generation (RAG) framework. This method uses a neural re-ranker to dynamically integrate trustworthy external knowledge sources like Wikipedia, ensuring that summaries are simplified, factually accurate, and contextually relevantLewis et al. (2020); Kang et al. (2024).

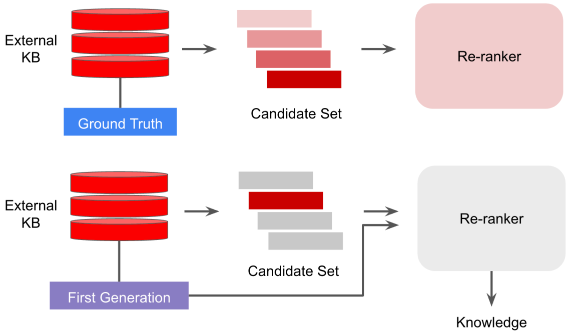
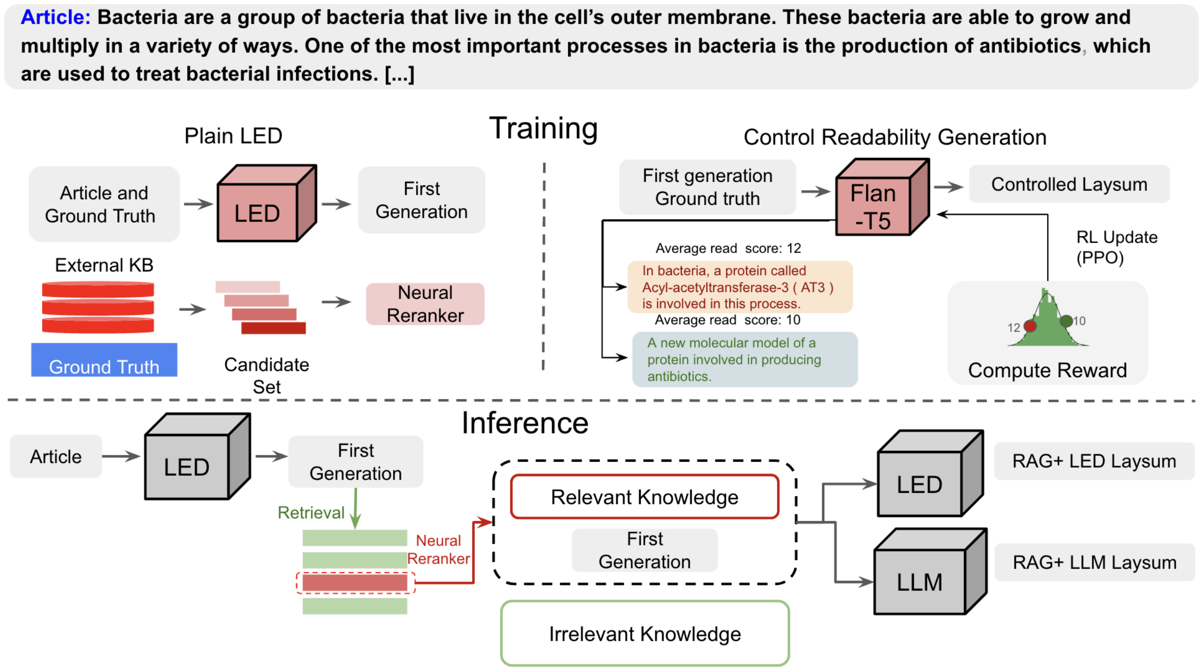


Figure 1: Knowledge Retrieval Augmented, with the trained re-ranker, can provide more relevant knowledge based on the first generation.

The architecture of the proposed RAG is illustrated in Figure 1. We also introduce a reward-based approach to overcome the limitations of traditional fine-tuning, which often produces summaries that have high ROUGE scores but are not actually readable to humans. This method fine-tunes the model by rewarding outputs that align with read-

[1] Our code and implementation details are available here: https://github.com/JoyDajunSpaceCraft/RAG-RLRC-LaySum

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 810–817
August 16, 2024. ©2024 Association for Computational Linguistics



Article: Bacteria are a group of bacteria that live in the cell's outer membrane. These bacteria are able to grow and multiply in a variety of ways. One of the most important processes in bacteria is the production of antibiotics, which are used to treat bacterial infections. [...]

Figure 2: This figure illustrates the architecture of the proposed RAG-RLRC-LaySum model. During the training phase, we employ the Longformer Encoder-Decoder (LED) model as the backbone Beltagy et al. (2020). We enhance the model's capabilities through Wikipedia knowledge retrieval during inference. We utilize large language models (LLMs) such as ChatGPT and Gemini to further improve readability and enhance textual clarity by modifying prompts. For controlled text generation, readability scores are utilized to guide the model in generating expected outputs. The outputs of these scores are normalized to ensure text consistency and quality across generated texts.

ability metrics (Flesch-Kincaid Grade Level and Dale-Chall Readability Score Foster and Rhoney (2002); Ribeiro et al. (2023)). Unlike traditional supervised methods that might limit the model's adaptability, our approach encourages the model to alternative expressions to enhance clarity. The BioLaySumm challenge is a research competition that focused on developing and benchmarking models for generating lay summaries from complex biomedical literature[2]. Our method is ranked 11th on the leaderboard of the challenge Goldsack et al. (2024).

2 Methodology

First, we employ a Retrieval-Augmented Generation (RAG) solution that processes entire papers despite limited input capacity. Second, we improve summary quality by optimizing readability using relevant background information. The framework is illustrated in Figure 2.

2.1 Retrieval Augmented Generation (RAG)

Our RAG framework enhances keyword-based retrieval by using an initial lay summary generated by the model as a query. In the inference stage, it retrieves relevant descriptions from Wikipedia Ponzetto and Strube (2007) by the Pyserini Lin et al. (2021) index. However, retrieving relevant information from a large number of articles remains a challenge because the first generated summaries cannot work as effective queries. We initially use the ground truth as a query but switch to the first generated layman summary during inference for passage retrieval. However, there's a risk that the top-k passages may not be the most relevant for generating accurate summaries. Given a scientific document D with a set of candidate passages $K = \{k_1, k_2, ..., k_n\}$ from grounding sources, the RAG framework generates a lay summary S by maximizing the probability:

$$p(S|D,K) = \prod_{i=1}^{|S|} p(s_i|s_{<i}, D, K) \qquad (1)$$

where s_i represents the i-th token in the summary, and $s_{<i}$ denotes the sequence of tokens preceding s_i. We use ColBERT Khattab and Zaharia (2020) and BGE-v2 Li et al. (2023a); Chen et al. (2024) as two different types of the neural re-ranker. The details about the trained re-ranker are in Ap-

[2] https://biolaysumm.org/

pendix B.

2.2 Reinforcement Learning for Readability Control (RLRC)

For details about the reranking model and sequence generation model training can be seen in Appendix A. The RLRC method inputs the first generation from the plain LED and uses the ground truth as the expected output. Our RLRC approach employs a reinforcement learning strategy to fine-tune the readability of summaries. We define a reward function $R(y, r^*)$ based on the desired readability level r^* that encourages the generation of text towards better readability, which is measured by the Flesch Reading Ease score r^*:

$$R(y, r^*) = 1 - \exp\left(-\frac{(R(y) - r^*)^2}{2\sigma^2}\right) \quad (2)$$

where $R(y)$ denotes the readability score of the generated summary y, and σ is a hyperparameter that controls the sensitivity of the reward function to deviations from the target readability score. Also, we leverage a Gaussian-based reward that strongly penalizes great variations in the readability Ribeiro et al. (2023).

We employ the Proximal Policy Optimization (PPO) Schulman et al. (2017) algorithm to optimize our RLRC model. The objective is to adjust the model's parameters θ by maximizing the objective function:

$$L(\theta) = \mathbb{E}_{(y,r^*)\sim p_{\theta_{\text{old}}}}\left[\left(\frac{p_\theta(y \mid D, r^*)}{p_{\theta_{\text{old}}}(y \mid D, r^*)}\right) R(y, r^*)\right] \quad (3)$$

Here, $p_{\theta_{old}}$and p_θ denote the policy under the old and current parameters, respectively.

2.3 Large Language Models

We use the LLMs in two ways: first, as a paraphrasing tool during inference to refine initial generations, and second, for directly generating layman summaries. This implementation is built on Gemini-1.0-pro, developed by Google Team et al. (2023), which also serves as our baseline LLM. We aim to create readable summaries while incorporating as many input keywords as possible. We follow Gemini-1.0-pro's default settings, and the prompt details are described in Appendix C.

3 Experimental Settings and Results

3.1 Datasets and Evaluation

This study uses biomedical research articles from the PLOS and eLife datasets, which include both technical abstracts and expert-crafted lay summaries. The PLOS dataset contains 24,773 training and 1,376 validation instances, while the eLife dataset comprises 4,346 training and 241 validation instances Goldsack et al. (2022). We assess summarization quality using predefined metrics: Relevance is gauged by ROUGE Scores (ROUGE-1, ROUGE-2, ROUGE-L) Lin (2004) and BERTScore Zhang et al. (2019); Readability by the Flesch-Kincaid Grade Level (FKGL) Kincaid et al. (1975), Dale-Chall Readability Score (DCRS) Dale (1948), and Learnable Evaluation Metric for Text Simplification (LENS) Maddela et al. (2022); Factuality by Summac Laban et al. (2022) and AlignScore Zha et al. (2023).

3.2 Performance of Baseline Models

The Plain LED model, serving as our baseline, achieved ROUGE-L scores of 41.33 and 44.07. In contrast, the Plain retrieve+LED model, which integrates external knowledge through the BM25 retriever, slightly improved ROUGE-L scores to 47.02 and 47.21. This indicates that the incorporation of external knowledge slightly enhances the relevance of the summaries.

3.3 Effect of Neural Re-rankers

Further improvements were observed with the RAG+LED model, which incorporates a trained neural re-ranker, boosting the ROUGE-L scores to 49.68 and 49.79. This significant increase demonstrates that neural re-rankers are more precise in selecting relevant content, effectively enhancing the accuracy and relevance of the summaries.

3.4 Effect of Large Language Models

The RAG+ChatGPT and RAG+Gemini models, utilizing LLMs, achieved high FKGL readability scores of 9.93 and 9.25 respectively, but their ROUGE-L scores were lower at 39.59 and 39.20, indicating that LLMs can sometimes introduce irrelevant information. Similarly, the Plain Gemini model, which relies solely on an LLM, scored only 33.63 in ROUGE-L, demonstrating the challenges LLMs face in producing coherent and accurate summaries without mechanisms for precise content selection.

3.5 Effect of RLRC

The RAG+RLRC model, integrating reinforcement learning training strategies, achieved a ROUGE-L score of 47.24. It marked an improvement in

Table 1: Results on PLOS and eLife validation datasets. For the ↑ means, the higher, the better; for the ↓ means, the lower, the better. All best results are marked as bold. The RAG+different models represent the models that used neural re-ranker.

Method	Relevance				Readability				Factuality	
	Rouge1↑	Rouge2↑	RougeL↑	BERTScore↑	FKGL↓	DCRS↓	CLI↓	LENS↑	AlignScore↑	SummaC↑
PLOS										
Plain LED	45.96	**15.00**	41.33	**85.97**	15.17	12.26	16.42	54.96	**81.68**	**74.34**
Plain retrieve+LED	45.53	14.37	41.10	85.66	15.06	12.10	16.32	51.82	77.38	71.94
RAG+LED	45.64	15.37	**42.30**	85.21	15.22	11.93	15.92	53.42	76.57	72.83
RAG+ChatGPT	37.39	6.81	33.96	84.70	**11.21**	**10.37**	**12.50**	71.90	65.71	57.85
RAG+Gemini	38.89	8.74	35.11	85.12	11.33	10.48	13.38	**74.76**	68.40	58.88
Plain Gemini	44.67	13.36	40.26	85.87	15.71	11.84	17.98	62.64	74.18	52.82
RAG-RLRC	**46.58**	14.96	41.81	85.83	14.89	11.81	16.78	47.55	78.45	72.97
eLife										
Plain LED	47.02	12.52	44.07	**84.73**	10.52	9.33	11.49	73.45	**62.37**	60.12
Plain retrieve+LED	47.72	12.40	44.26	84.41	12.11	9.25	11.40	67.57	53.57	56.18
RAG+LED	47.69	12.41	44.34	84.41	11.99	9.25	**11.39**	67.95	53.89	55.65
RAG+ChatGPT	39.78	7.23	37.13	84.02	**9.58**	9.49	11.40	75.40	58.96	50.44
RAG+Gemini	39.90	9.04	36.97	84.29	**9.58**	9.65	12.47	**78.93**	62.91	55.81
Plain Gemini	22.60	3.22	20.85	80.81	16.38	12.72	24.18	52.44	53.19	44.97
RAG-RLRC	**47.91**	**12.65**	**44.96**	84.61	10.52	**9.11**	11.73	68.61	61.34	**60.40**
Average										
Plain LED	46.49	13.76	42.70	**85.35**	12.84	10.79	13.95	64.20	**72.02**	**67.23**
Plain retrieve+LED	46.62	13.38	42.68	85.03	13.58	10.67	13.86	59.69	65.47	64.06
RAG+LED	46.66	**13.89**	43.32	84.81	13.60	10.59	13.65	60.68	65.23	64.24
RAG+ChatGPT	38.59	7.02	35.55	84.36	**10.40**	**9.93**	**11.95**	73.65	62.34	54.15
RAG+Gemini	39.39	8.89	36.04	84.70	10.46	10.06	12.93	**76.85**	65.66	57.35
Plain Gemini	33.63	8.29	30.55	83.34	16.04	12.28	21.08	57.54	63.68	48.89
RAG-RLRC	**47.24**	13.80	**43.38**	85.22	12.70	10.46	14.25	58.08	69.89	66.68

factual accuracy, with a Summac score of 78.45 compared to the 73.44 of Plain LED. This highlights the effectiveness of reinforcement learning strategies in optimizing the text's factual alignment.

4 Related Work

Automatic summarization in the biomedical domain has been extensively studied Du et al. (2019); Krishna et al. (2020); Goldsack et al. (2023a); Devaraj et al. (2022). The primary challenge in this field is simplifying the content of original articles to make them comprehensible to laypersons. While Rosati (2023) supplement source documents to aid in generating more comprehensible summaries, and Devaraj et al. (2022) explore how text simplification impacts summary accuracy, introducing a taxonomy of error types and identifying omissions as a prevalent issue, these approaches often overlook the balance between simplification and factuality.

To enhance summary factuality, researchers incorporate factual knowledge from external sources during model training Mao et al. (2022), which has proven effective in improving accuracy. Rosati (2023) utilize Wikipedia to enrich summaries with additional knowledge, while Poornash et al. (2023) employ a trained re-ranker to select pertinent information, enhancing the factuality of summaries.

5 Conclusion and Future Work

The RAG-RLRC-LaySum framework effectively simplifies complex biomedical texts, enhancing readability and factual accuracy for lay audiences. It surpasses traditional models, offering new insights into the pivotal role of knowledge retrieval and readability optimization in scientific summarization. Future work will expand the framework's knowledge sources and refine how knowledge is utilized, potentially broadening its application across various scientific fields. This will further explore the integration of domain-specific knowledge to improve the precision and relevance of summaries.

6 Limitations

While the RAG-RLRC-LaySum framework shows promise, it has several limitations. The reliance on external sources like Wikipedia can introduce biases. The framework's computational complexity is high, making real-time applications challenging. Readability metrics like FKGL and DCRS may not fully capture readability for all audiences. Additionally, the generalizability to other domains beyond biomedical texts is uncertain. Lastly, evaluations based on automated metrics may not fully reflect user experience, highlighting the need for human evaluations. Future work should address these limitations by exploring diverse knowledge sources, optimizing efficiency, refining readability metrics, and conducting human evaluations.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *Preprint*, arXiv:2402.03216.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

D Dale. 1948. The dale-chall formula for predicting readability. *Educational Research Bulletin*, 27:11–20.

Ashwin Devaraj, William Sheffield, Byron C Wallace, and Junyi Jessy Li. 2022. Evaluating factuality in text simplification. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2022, page 7331. NIH Public Access.

Nan Du, Kai Chen, Anjuli Kannan, Linh Tran, Yuhui Chen, and Izhak Shafran. 2019. Extracting symptoms and their status from clinical conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 915–925, Florence, Italy. Association for Computational Linguistics.

David R Foster and Denise H Rhoney. 2002. Readability of printed patient information for epileptic patients. *Annals of Pharmacotherapy*, 36(12):1856–1861.

Saadia Gabriel, Asli Celikyilmaz, Rahul Jha, Yejin Choi, and Jianfeng Gao. 2020. Go figure: A meta evaluation of factuality in summarization. *arXiv preprint arXiv:2010.12834*.

Tomas Goldsack, Zheheng Luo, Qianqian Xie, Carolina Scarton, Matthew Shardlow, Sophia Ananiadou, and Chenghua Lin. 2023a. Overview of the biolaysumm 2023 shared task on lay summarization of biomedical research articles. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 468–477, Toronto, Canada. Association for Computational Linguistics.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chen Tang, Carolina Scarton, and Chenghua Lin. 2023b. Enhancing biomedical lay summarisation with external knowledge graphs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8016–8032, Singapore. Association for Computational Linguistics.

Minki Kang, Seanie Lee, Jinheon Baek, Kenji Kawaguchi, and Sung Ju Hwang. 2024. Knowledge-augmented reasoning distillation for small language models in knowledge-intensive tasks. *Advances in Neural Information Processing Systems*, 36.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48.

Peter Kincaid, Robert P. Fishburne, Richard L. Rogers, and Brad S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Kundan Krishna, Sopan Khosla, Jeffrey P. Bigham, and Zachary C. Lipton. 2020. Generating SOAP notes from doctor-patient conversations. *CoRR*, abs/2005.01795.

Philippe Laban, Tobias Schnabel, Paul N Bennett, and Marti A Hearst. 2022. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Chaofan Li, Zheng Liu, Shitao Xiao, and Yingxia Shao. 2023a. Making large language models a better foundation for dense retrieval. *Preprint*, arXiv:2312.15503.

Panfeng Li, Mohamed Abouelenien, and Rada Mihalcea. 2023b. Deception detection from linguistic and physiological data streams using bimodal convolutional neural networks. *arXiv preprint arXiv:2311.10944*.

Panfeng Li, Qikai Yang, Xieming Geng, Wenjing Zhou, Zhicheng Ding, and Yi Nian. 2024. Exploring diverse methods in visual question answering. *arXiv preprint arXiv:2404.13565*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: An easy-to-use python toolkit to support replicable ir research with sparse and dense representations. *arXiv preprint arXiv:2102.10073*.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2022. Lens: A learnable evaluation metric for text simplification. *arXiv preprint arXiv:2212.09739*.

Qianren Mao, Jianxin Li, Hao Peng, Shizhu He, Lihong Wang, Philip S. Yu, and Zheng Wang. 2022. Fact-driven abstractive summarization by utilizing multi-granular multi-relational knowledge. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:1665–1678.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. *arXiv preprint arXiv:2005.00661*.

Simone Paolo Ponzetto and Michael Strube. 2007. An api for measuring the relatedness of words in wikipedia. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 49–52.

AS Poornash, Atharva Deshmukh, Archit Sharma, and Sriparna Saha. 2023. Aptsumm at biolaysumm task 1: Biomedical breakdown, improving readability by relevancy based selection. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 579–585.

Leonardo F. R. Ribeiro, Mohit Bansal, and Markus Dreyer. 2023. Generating summaries with controllable readability levels. In *Conference on Empirical Methods in Natural Language Processing*.

Domenic Rosati. 2023. Grasum at biolaysumm task 1: Background knowledge grounding for readable, relevant, and factual biomedical lay summaries. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 483–490.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. 2022. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*.

Fali Wang, Runxue Bao, Suhang Wang, Wenchao Yu, Yanchi Liu, Wei Cheng, and Haifeng

Chen. 2024. Infuserki: Enhancing large language models with knowledge graphs via infuser-guided knowledge integration. *arXiv preprint arXiv:2402.11441*.

Qikai Yang, Panfeng Li, Zhicheng Ding, Wenjing Zhou, Yi Nian, and Xinhe Xu. 2024. A comparative study on enhancing prediction in social network advertisement through data augmentation. *arXiv preprint arXiv:2404.13812*.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2022. Linkbert: Pretraining language models with document links. *arXiv preprint arXiv:2203.15827*.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *arXiv preprint arXiv:2305.16739*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

A Finetuning models

For training the Longformer Encoder-Decoder (LED) model Beltagy et al. (2020), we utilized the "allenai/led-base-16384" pre-trained checkpoint available on Huggingface's model hub. Our training setup included a configuration that processes 16,384 input tokens and generates outputs limited to 512 tokens. This training was conducted over the course of a single epoch.

In parallel, we employed BioLinkBERT-base Yasunaga et al. (2022) as the foundational language model for processing eLife and PLOS datasets, leveraging its specialized capabilities in understanding biomedical context.

Then, we designed a neural re-ranker based on the ColBERT Khattab and Zaharia (2020) and BGE-v2 Li et al. (2023a); Chen et al. (2024) scoring mechanism, which refines the results by evaluating the relevance of retrieved documents. The training for this re-ranker was tailored to accept inputs of up to 512 tokens, and it was fine-tuned to generate models by considering the top 5 most pertinent retrieval results. Futhermore, we define the Flan-T5-Large from huggingface, we use the model "google/flan-t5-large" as the base model. To make use of the control generation, we use the keywords in the article as the expected output to make sure the relevance.

A.1 Retrieval Augmented Generation

We conduct the experiment based on the model Longformer Encoder-Decoder (LED) Beltagy et al. (2020) which supports an input token length of 16,384 tokens. For the basic fine-tuning method, we find out in both the PLOS and eLife data that the re-ranker result will be a higher result in the Rouge-L and a lower score in the FKGL and DCRS score. In that case, indicate the lower the complexity of the description.

We use ColBERT Khattab and Zaharia (2020) and BGE-v2 Li et al. (2023a); Chen et al. (2024) as two different types of the neural re-ranker.

A.2 Reinforcement Learning for Readability Control (RLRC)

By utilizing various control levels for readability within the model-generated results, we focus on understanding how modifications to the readability scores, particularly the Flesch-Kincaid Grade Level (FKGL), impact the final summaries. The Flan-T5 model Chung et al. (2024) serves as the primary backbone for text generation. During the inference phase on testing data, where no ground truth is available for the reward mechanism, keywords are used as proxy indicators to ensure that the generated summaries accurately reflect the expected concepts.

In our model, we define two key mathematical expressions. The first is the Gaussian probability density function, used to estimate the likelihood of a given value within a normal distribution. The expression for this function is:

$$\begin{aligned}&\text{calc\_nd}(value, mean) = \\ &\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(value - mean)^2}{2\sigma^2}\right)\end{aligned} \tag{4}$$

This function is essential for assessing how far a data point deviates from the mean and is widely used in statistical analyses.

The second formula defines our reward function, which combines three different scoring metrics—readability score, BERTScore, and text length score—to comprehensively evaluate the quality of the text. The formula is as follows:

$$\begin{aligned}\text{reward} &= w_r \cdot \text{normalized\_flesch\_scores} + \\ &w_b \cdot \text{all\_bertscore\_scores} + w_l \cdot \text{length\_scores}\end{aligned} \tag{5}$$

Here, w_r, w_b and w_l are the weight factors for each scoring metric, adjusting the influence of each score in the overall assessment. By default, we set $w_r = 0.5, w_b = 0.3, w_l = 0.2$.

This weighted approach allows us to tailor the scoring criteria to different types of text analysis tasks, accommodating the multifaceted nature of text data.

B Retrieval Design

For the reranking of retrieved documents, we utilize the pyserini package Lin et al. (2021). Following the approach outlined by Rosati (2023), we employ $enwiki-paragraphs$ for background knowledge. We first retrieved 20 candidate paragraphs and then rerank the top 5 results.

B.1 Neural Re-ranker

In the provided Table 2, the performance trends across the eLife and PLOS datasets reveal that neural re-ranking methods (ColBERT and BGE) consistently outperform the traditional BM25 method. Notably, BGE shows a clear upward trend in accuracy from Top1 through Top20 in both datasets. Similarly, ColBERT's performance also exhibits an upward trajectory, although it remains below BGE, indicating a strong but second-tier efficacy among the tested methods.

Table 2: Accuracy for Neural Re-ranker.

Dataset	Method	Top1	Top5	Top20
eLife	BM25	10.32	42.13	65.24
	ColBERT	15.38	53.85	76.92
	BGE	18.53	60.19	78.52
PLOS	BM25	20.33	53.74	80.12
	ColBERT	26.09	57.97	84.06
	BGE	29.30	59.98	88.92

C Prompts

Table 3: One shot prompt for ChatGPT 4 and Gemini 1.0.

System: You are a layman rephrase; your goal is to rephrase the input and make it easier to read. For example: 'Diabetes is a condition in which the pancreas cannot produce enough insulin to feed the body. This is caused by a protein called proinsulin is an ingredient a group of molecules called cysteine thiols. The rephrased result should be: 'Diabetes is a condition where the pancreas doesn't produce enough insulin to meet the body's needs. This happens because of a protein called proinsulin, which consists of a group of molecules known as cysteine thiols.'

Input: Here is the original text I want you to help me to rephrase: {first generation}. Make it easier to read and retain as much of the biomedical phrase as possible and have a similar length as the original text.

Table 4: Prompt used for Gemini for article summarization.

I will give you a long article in biomedical publications, you should generate an abstractive summarization of this article in one single paragraph. I will also give you the keyphrases in this article, you should try to include as many keyphrases in your generated summarization as possible. The summarization is with an emphasis on catering to non-expert audiences through the generation of summaries that are more readable, containing more background information and less technical terminology. Keyphrases:{}, Article:{}.

Team YXZ at BioLaySumm: Adapting Large Language Models for Biomedical Lay Summarization

Jieli Zhou[1*], Cheng Ye[2*], Pengcheng Xu[3], and Hongyi Xin[1]

[1]UM-SJTU Joint Institute, Shanghai Jiao Tong University
[2]Zhejiang Laboratory
[3]University of Illinois Urbana-Champaign
[*]These authors contribute equally to this work.
Correspondence: zhoujieli@sjtu.edu.cn, hongyi.xin@sjtu.edu.cn

Abstract

Biomedical literature are crucial for disseminating new scientific findings. However, the complexity of these research articles often leads to misinterpretations by the public. To address this urgent issue, we participated in the BioLaySumm task at the 2024 ACL BioNLP workshop, which focuses on automatically simplifying technical biomedical articles for non-technical audiences. We conduct a systematic evaluation of the SOTA large language models (LLMs) in 2024 and found that LLMs can generally achieve better readability scores than smaller models like Bart. Then we iteratively developed techniques of title infusing, K-shot prompting, LLM rewriting and instruction finetuning to further boost readability while balancing factuality and relevance. Notably, our submission achieved the first place in readability at the workshop, and among the top-3 teams with the highest readability scores, we have the best overall rank. Here, we present our experiments and findings on how to effectively adapt LLMs for automatic biomedical lay summarization. Our code is available at `https://github.com/zhoujieli/biolaysumm`.

1 Introduction

The biomedical literature is one of the most important information sources for researchers to share their latest discoveries. However, the increasing volume of information has become overwhelming, e.g. PubMed alone hosts over 36 million papers, with more than one million new articles added annually (Jin et al., 2024). This information deluge makes it challenging for even specialized biomedical experts to keep up with the latest research, let alone the general public. General public, although having a keen interest in biomedical research due to its relevance to everyday life, may be prohibited by the difficult biomedical terminology, experimental setups, or the metric abbreviations. Currently, media outlets play a crucial role in bridging the gap between scientific literature and public understanding (Peters, 2013). However, media often lack the necessary context when reporting new studies, which can lead to exaggerated claims. For instance, reports may claim that certain diets promote longevity[1], but closer inspection of the literature reveals that these claims are typically based on preliminary animal studies and lack robust support in human studies (Bruijnis et al., 2013; Janssen et al., 2019; Murphy et al., 2014).

In recent years, with the rapid advancements of Transformer-based natural language processing algorithms, many intelligent systems have been developed to automate the process of explaining and summarizing research papers to lay people. Dangovski et al. (2021) collected 100,000 webpages from Science Daily, a popular press release websites for research papers, and finetuned a Bert model to produce automatic science journalism; Cohan et al. (2018) developed a hierarchical disource-aware attention model to effectively summarize the long and hierarchical research paper. Interactive research paper summarization systems like SciSummary [2], Scholarcy[3] and SciSpace[4] can help researchers quickly get the gist from the literature and develop a map of connected research (Nahas, 2024).

In particular, biomedical literature contains many domain jargons which lack readable explanations; background stories are frequently omitted, and findings are hidden in obscure metrics names without direct discussions. In essense, the literature cannot "talk" by itself and answer general public's questions. To advance the research in automatic lay summary generation for biomedical literature, the ACL BioNLP task of BioLaySumm was started

[1] `https://www.hsph.harvard.edu/nutritionsource/media/`

[2] `https://scisummary.com/`

[3] `https://www.scholarcy.com/`

[4] `https://typeset.io/`

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 818–825
August 16, 2024. ©2024 Association for Computational Linguistics

in 2023 (Goldsack et al., 2023, 2024). Accompanying this task, a lay summary dataset was curated to facilitate model training and evaluation (Goldsack et al., 2022). The dataset contains 31,020 article-summary pairs[5] from two journal series, Public Library of Science (PLOS) and eLife. Each article-summary pair contains the paper full text and the corresponding lay summaries written by paper authors (PLOS) or journal editors (eLife). Due to the different sources of lay summaries, the characteritics of these summaries vary, e.g. PLoS lay summaries are on average 175.6 words and eLife summaries are on average 347.6 words [6]. The task of BioLaySumm is then given the full article text, a system should automatically output a lay summary, which will be measured in 10 metrics of factuality, readability and relevance.

In the previous iteration of BioLaySumm, we observe that the best performing teams used advanced large language models (LLMs) like GPT-3.5 to produce zero-shot lay summaries (Turbitt et al., 2023) and augment training data (Sim et al., 2023). In addition, we have frequently used tools like GPT4, `Kimi.ai`, and `ChatDoc.com` to facilitate rapid understanding of research papers. In this work, we make several key contributions. Firstly, we conduct a comprehensive set of experiments demonstrating that while directly prompting Large Language Models (LLMs) can enhance the readability of lay summaries, it often compromises their factuality and relevance. Secondly, we have developed a suite of adaptation techniques, including **title infusion**, **K-shot prompting**, **LLM rewriting**, and **instruction fine-tuning**, which enable LLMs to generate lay summaries that are well-balanced in terms of factuality, relevance, and readability. Notably, our approach achieved first place in readability in the 2024 BioLaySumm competition, while also maintaining a strong balance across the other evaluated metrics.

2 Dataset and Evaluation Metrics

The dataset (Goldsack et al., 2022) of 31,020 article-summary pairs is divided into three parts, train, validation and test, where the train and validation sets are given for model development, and the lay summaries of the test set are hidden. The distribution of the data is shown in Table 1. Model outputs are submitted to a competition website and the scores are computed in around 1.5 hours.

Table 1: **Distribution of the BioLaySumm data**

Dataset	Train	Val	Test	# words
PLOS	24,773	1,376	142	175.6
eLife	4,346	241	142	347.6

Model-generated lay summaries are evaluated against 10 metrics of three categories, relevance, readability and factuality.

Relevance: ROUGE or Recall-Oriented Understudy for Gisting Evaluation (1, 2, and L) (Lin, 2004) and **BERTScore** (Zhang et al.) are the relevance metrics which uses lexical or embedding based methods to measure the overlap between the generated summaries to the reference ones. The higher the scores the more relevant the summaries.

Readability: Flesch-Kincaid Grade Level (**FKGL**)(Kincaid et al., 1975), Dale-Chall Readability Score (**DCRS**)(Chall and Dale, 1995), Coleman-Liau Index (**CLI**)(Coleman and Liau, 1975) are lightweight readability metrics that predict the US grade level of education needed to understand the generated summaries, so the lower the scores, the more readable the summaries. Whereas **LENS** (Learnable Evaluation Metric for Simplification) (Maddela et al., 2022), a readability metric trained on human judgment data, aligns more closely with human preferences, the goal is to achieve a higher LENS score.

Factuality: Alignscore (Zha et al., 2023) is an automatic factual consistency metric for checking whether all information in the summary is contained in the reference. Similarly, **SummaC** (Laban et al., 2022) or Summary Consistency is a natural language inference (NLI) method that measures the factual consistency between generated summary and reference sentence-wise. The goal is to maximize these two factuality metrics.

3 Method and Results

Following previous work (Turbitt et al., 2023), we mainly focus on using LLMs in this work. In our settings, the input is the article text and relevant metadata, and different LLMs are prompted along with a system prompt to generate the lay summary. In this section, we explain our methods and the results in detail.

[5] https://biolaysumm.org/

[6] https://aclanthology.org/2023.bionlp-1.44.mp4

Table 2: **Large Language Models' Performances on BioLaySum test set**. 5 SOTA LLMs are prompted to generate lay summarization for the test dataset, the title for each article is retrieved and added to the prompt. Overall, LLM models, although lacking in relevance and factuality, generally outperform the baseline model in readability metrics. Notice that the Llama3 models beat the baseline by a large margin in the LENS metric.

Method	Relevance				Readability				Factuality	
	R1 ↑	R2 ↑	RL ↑	BS ↑	FKGL↓	DCRS↓	CLI↓	LENS↑	AS ↑	SC ↑
Baseline (BART)	**0.4696**	**0.1395**	**0.4358**	**0.8623**	12.0359	10.1475	13.4852	48.0963	**0.7788**	**0.7026**
Claude-3-Opus	0.4426	0.1194	0.3966	0.8506	13.2162	10.1755	15.2807	73.093	0.5794	0.4912
Gemini-1.5-pro	0.4405	0.11	0.4076	0.8554	13.6968	10.4408	16.0192	73.5596	0.6823	0.5004
GPT-4	0.4299	0.0983	0.3837	0.8524	14.362	10.7441	15.9209	72.2514	0.5818	0.452
Llama3-8B-Instruction	0.4152	0.1065	0.3847	0.854	11.6099	**9.2043**	**12.8627**	80.1454	0.6539	0.5172
OpenBioLLM-Llama3-70B	0.4104	0.0993	0.3801	0.855	**11.0162**	9.369	12.9965	**81.2052**	0.7018	0.5463

3.1 Title Infusing

We observe that article titles are missing in the test data. However, we think titles are essential for summarization since it encapsulates the high-level description of the article by the authors. To retrieve the title for the articles, we used **BeautifulSoup4**[7] to retrieve the article titles from the eLife and PLOS website based on the DOI urls. The titles are then infused into the prompt for LLMs to better position its lay summarization. All models in this work have the information of the title at inference time.

3.2 Large Language Models

Since the last iteration of BioLaySumm, more advanced LLMs emerged which showcased better reasoning and text processing skills. We benchmarked five SOTA LLMs against the official baseline method based on BART: Anthropic's Claude-3-Opus[8], Google's Gemini-1.5-Pro[9], OpenAI's GPT-4 [10] and Meta's Llama-3 [11]. These LLMs have over billions or even hundreds of billions of parameters and are very good at instruction following. We also included, OpenBioLLM-LLama3-70B (Ankit Pal, 2024), which is a Llama-3-70B model finetuned on biomedical domain and is reported to specialize in various BioNLP tasks. Due to the prohibitive costs, we limit the input tokens to only the abstract part of the test set. The result of these 5 LLMs on test data is shown in Table 2. It is surprising that even the most advanced LLMs cannot surpass the Bart-baseline model in terms of relevance or factuality. However, it is evident that LLMs hold a distinct advantage in readability, particularly as measured by the LENS score. Next, building on these findings, we aimed to enhance readability while also improving the metrics for relevance and factuality.

3.3 Finetuning for Relevance and Factuality

LLMs are autoregressive transformer models which are trained to predict the next token[12]. Without further fine-tuning, they lack capabilities in producing factual and relevant summaries, as shown in table 2. Inspired by the success of techniques like Supervised Fine-Tuning (SFT) (Alt et al., 2019) and instruction tuning (IT) (Wu et al., 2024) on LLMs in solving downstream NLP tasks, we adopt a parameter efficient finetuning technique called Low-Rank Adaption (LoRA) (Hu et al., 2021) to further fine-tune two models, one for PLOS and another for eLife, due to the different characteristics of these two journal lay summaries as shown in table 1. We first construct instruction tuning data from the article-summary pair by including an instruction prompt, **"Write lay summary for the given input (a summary that is suitable for non-experts). Here is the article:..."**. We use the article full text (up to 8K as of the Llama3's context window) as input, and the corresponding lay summary as output. We report the result on the validation dataset in Table 3, and for the test data submission, we perform instruction fine-tuning on the entire train-val dataset and predict on the test dataset. The experiments were done on a GPU server with 8 NVIDIA RTX 4090Ti 48GB GPUs. We used Llama-3-8B-Instruct [13] as our base model,

[7] https://beautiful-soup-4.readthedocs.io/en/latest/

[8] https://www.anthropic.com/api

[9] https://blog.google/technology/ai/google-gemini-next-generation-model-february-2024/

[10] https://openai.com/index/gpt-4/

[11] https://llama.meta.com/llama3/

[12] https://huggingface.co/docs/transformers/llm_tutorial

[13] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

Table 3: Performance of Llama-3-8B model on the validation dataset with and without Instruction-finetuning. With finetuning the metrics of 8B model compare favorably to larger un-finetuned 70B model, and is consistently better than the baseline Llama3-8B model in relevance and factuality.

Method	Relevance				Readability				Factuality	
	R1 ↑	R2 ↑	RL ↑	BS ↑	FKGL↓	DCRS↓	CLI↓	LENS↑	AS ↑	SC ↑
Llama3-8B	0.4185	0.1049	0.3855	0.8563	11.7285	9.2582	**12.9303**	80.1033	0.6316	0.5282
Llama3-8B-FT	**0.4297**	**0.111**	**0.3977**	0.8513	11.8257	**8.6854**	13.1627	**80.7012**	**0.70628**	**0.5816**
OpenBioLLM-70B	0.4180	0.1071	0.3821	**0.8583**	**11.6109**	9.8948	13.3855	80.0218	0.6966	0.5763

and training goes on for 3 epochs, with learning rate of 5e-5. More details are available in our code repo.

3.4 K-Shot Prompting for Factuality

In constructing the prompts for LLMs, we used in-context learning (Dong et al., 2022) or few-shot learning [14], in which we provide some examples to the LLMs. For test data, we proposed to use the top K semantically similar articles in the train and validation dataset or K-shot prompting. To compute the semantic similarities, we used a recent state-of-the-art embedding model BGE M3 (Chen et al., 2024) from BAAI[15] to compute and pick the K most similar abstracts from the train-val dataset to the given test article. Together with the input article, the K-Shot example pairs are sent to the LLMs. Due to time limit, we set K=1. We experimented this K-shot Prompting technique with the smaller Llama-3-8B model, and observe better performance metrics in factuality as shown in Table 4. This implies LLMs can be prompted with semantically similar lay summary example to be more grounded in the origincal text, boosting factuality.

Table 4: Factuality Scores for Llama3-8B-Instruction Models with and without K-shot Prompting on the test dataset, K=1

Model	AlignScore↑	SummaC↑
Llama3-8B (kshot)	**0.7523**	**0.5582**
Llama3-8B	0.6539	0.5172

3.5 LLM-rewrite for Readability

In observing the results from the five LLMs as in table 2, we hypothesized that Bart model finetuned on the given dataset does better in relevance and factuality while LLMs may be better in readability. We aim to boost the readability of Bart, thus confirming our hypothesis that LLMs' lay summaries are better in readability, so we developed a strategy of LLM-rewrite, where we first finetuned a Bart model, then used a specialized biomedical LLM, OpenBioLLM-LLama3-70B to rewrite the summary. It is observed that readability of the LLM-rewrote Bart summary is improved and especially in the LENS metric as shown in Table 5. However, Bart+rewrite still have consistently lower readability compared to LLM (OpenBioLLM-70B).

Table 5: Readability Metrics for BART-Finetuned Models with and without rewrite with OpenBioLLM-LLama3-80B.

Model	FKGL↓	DCRS↓	CL↓	LENS↑
BART	12.5053	9.948	13.5215	60.1214
BART+rewrite	11.1444	9.9033	13.4803	80.3856
LLM-70B	**11.0162**	**9.369**	**12.9965**	**81.2052**

4 Results and Conclusion

We constructed our final submission based a combination of three techniques: title infusing, instruction finetuning and K-Shot prompting. Overall, our submission achieved the 1st place in the readability catorgy, and had the better overall score compared with the top-3 team in the readability category (29th vs. 36th and 40th)

Team	**Relevance**	**Readability**	**Factuality**	**Overall**
YXZ	0.6845	0.8395	0.3190	29th
NLPSucks	0.3870	0.8297	0.5299	36th
jimmyapples	0.7008	0.8270	0.1875	40th

In this work, through systematic experiments on the capabilities and limitatons of the SOTA Large Language Models, we developed strategies to adapt LLMs for the task of BioLaySumm, and achieved the best result in readability while balancing other metrics. We provide experiment details and findings for researchers to keep advancing in this field.

[14] https://www.promptingguide.ai/techniques/fewshot

[15] https://www.baai.ac.cn/english.html

Limitations

Our study demonstrates effective strategies for adapting large language models (LLMs) to biomedical lay summarization, achieving good performances in readability while balancing factuality and relevance. However, several limitations warrant attention. First, while our techniques, title infusion, K-shot prompting, instruction tuning, and LLM rewriting showed promising results on the BioLaySumm datasets, their applicability to other types of biomedical papers, such as systematic reviews, remains untested. These methods may require further domain-specific adaptations for them to work well on other datasets. Second, the computational demands of current state-of-the-art LLMs are high, restricting their use in resource-limited settings. We will explore techniques to reduce model sizes and enable them for low-resource scenarios. Third, our efforts to balance readability with factuality and relevance reveal inherent tradeoffs, that is enhancing readability may sometimes oversimplify complex biomedical concepts, risking factual accuracy and detail omission. We plan to develop more balanced summarization strategies in future studies. Lastly, the ethical implications of using LLMs for generating lay summaries in highly sensitive biomedical fields are significant, especially given the risk of misinformation due to LLMs' hallucination issues. We will develop methods to automatically detect the toxic contents in the LLM outputs, and develop effective methods to correct them. In conclusion, while our study advances the field of LLM-based biomedical summarization, ongoing efforts are necessary to address these limitations and enhance the reliability and scope of our methodologies.

References

Christoph Alt, Marc Hübner, and Leonhard Hennig. 2019. Fine-tuning pre-trained transformer language models to distantly supervised relation extraction. *arXiv preprint arXiv:1906.08646*.

Malaikannan Sankarasubbu Ankit Pal. 2024. Openbiollms: Advancing open-source large language models for healthcare and life sciences. https://huggingface.co/aaditya/OpenBioLLM-Llama3-70B.

MRN Bruijnis, FLB Meijboom, and EN Stassen. 2013. Longevity as an animal welfare issue applied to the case of foot disorders in dairy cattle. *Journal of Agricultural and Environmental Ethics*, 26:191–205.

Jeanne Sternlicht Chall and Edgar Dale. 1995. Readability revisited: The new dale-chall readability formula. *(No Title)*.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *arXiv preprint arXiv:1804.05685*.

Meri Coleman and Ta Lin Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283.

Rumen Dangovski, Michelle Shen, Dawson Byrd, Li Jing, Desislava Tsvetkova, Preslav Nakov, and Marin Soljačić. 2021. We can explain your research in layman's terms: Towards automating science journalism at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12728–12737.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*.

Tomas Goldsack, Zheheng Luo, Qianqian Xie, Carolina Scarton, Matthew Shardlow, Sophia Ananiadou, and Chenghua Lin. 2023. Overview of the biolaysumm 2023 shared task on lay summarization of biomedical research articles. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Toronto, Canada. Association for Computational Linguistics.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Lieneke K Janssen, Nadine Herzog, Maria Waltmann, Nora Breuer, Kathleen Wiencke, Franziska Rausch, Hendrik Hartmann, Maria Poessel, and Annette Horstmann. 2019. Lost in translation? on the need for convergence in animal and human studies on the role of dopamine in diet-induced obesity. *Current addiction reports*, 6:229–257.

Qiao Jin, Robert Leaman, and Zhiyong Lu. 2024. Pubmed and beyond: biomedical literature search in the age of artificial intelligence. *Ebiomedicine*, 100.

J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Philippe Laban, Tobias Schnabel, Paul N Bennett, and Marti A Hearst. 2022. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2022. Lens: A learnable evaluation metric for text simplification. *arXiv preprint arXiv:2212.09739*.

Tytus Murphy, Gisele Pereira Dias, Sandrine Thuret, et al. 2014. Effects of diet on brain plasticity in animal and human studies: mind the gap. *Neural plasticity*, 2014.

Kamal Nahas. 2024. Is ai ready to mass-produce lay summaries of research articles? *Nature*.

Hans Peter Peters. 2013. Gap between science and media revisited: Scientists as public communicators. *Proceedings of the National Academy of Sciences*, 110(supplement_3):14102–14109.

Mong Yuan Sim, Xiang Dai, Maciej Rybinski, and Sarvnaz Karimi. 2023. Csiro data61 team at biolaysumm task 1: Lay summarisation of biomedical research articles using generative models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 629–635.

Oisín Turbitt, Robert Bevan, and Mouhamad Aboshokor. 2023. Mdc at biolaysumm task 1: Evaluating gpt models for biomedical lay summarization. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 611–619.

Zhengxuan Wu, Atticus Geiger, Thomas Icard, Christopher Potts, and Noah Goodman. 2024. Interpretability at scale: Identifying causal mechanisms in alpaca. *Advances in Neural Information Processing Systems*, 36.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *arXiv preprint arXiv:2305.16739*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

A Appendix

A.1 LLMs

In total, we spent around 200 USD on prompting various LLMs to generate lay summaries. For Claude-3-opus, we used API from Antropic, and for GPT-4, we used API from OpenAI. For Gemini-1.5-pro we used API from Google. Due to the prohibitive costs, we only tested the LLMs' performances on the test dataset.

For the local LLM models, BART, Llama3-8B-Instruction and OpenBioLLM-Llama3-70B, we downloaded the weights and checkpoints from Huggingface[16] and Modelscope[17]. All experiments were done on a single GPU-server with 8 NVIDIA RTX 4090 GPUs. We also used the Google Colab platform [18] for ideation and prototyping.

A.2 Prompts for LLMs

Our system prompt is set to be

System Prompt: "Please write a corresponding lay summary based on the content of the article provided. Requirements:
1. Lay summary needs to be easy to understand so that it can be quickly understood by non-specialists;
2. Lay summary needs to grasp the point of the article and be concise and to the point;
3. Just output a lay summary, no other content is required."

Then, the computed K-Shot example pairs are added along with the input (abstract/full text). An example of K-shot prompt is shown as below.

```
messages = [
{"role": "system",
"content": system_prompt},
{"role": "user",
"content": "Abstract: xxx"},
{"role": "assistant",
"content": "Lay summary: xxx"},
{"role": "user",
"content": "Abstract: xxx"},
```

[16] https://huggingface.co/

[17] https://www.modelscope.cn/

[18] https://colab.research.google.com/

```
{"role": "assistant",
 "content": "Lay summary: xxx"},
{"role": "user",
 "content": "Abstract: xxx"},
{"role": "assistant",
 "content": "Lay summary: xxx"},
{"role": "user",
 "content": "Abstract: {xxx}"}
 ]
```

A.3 Llama-3 Instruction Tuning

We construct instruction tuning dataset based on the given train and validation dataset, an example of the instruction tuning data is shown as below.

```
{
  "instruction": "Write lay summary
  for the given input (a summary that
  is suitable for non-experts).
  Here is the article.",
  "input":  article,
  "output": lay summary
}
```

Then we used unsloth [19] and LoRA [20] to conduct instruction finetuning. Code implementation is made available in our code repo https://github.com/zhoujieli/biolaysumm.

A.4 Analysis on Readability

To further corroborate our findings on readability, we evaluated local LLMs on the validation dataset and analyze their readability performances over the two journal datasets. The results are shown in Figure 1 and Figure 2. This is in line with our findings, and highlights a novel observation: lay summaries written by journal editors in eLife have better and harder-to-beat readability whereas in PLOS, authors' own lay summaries are generally worse in readabilities. This observation provides a unique opportunity for lay summarization systems like ours to help in generating lay summaries for the authors, and make their research more readable.

[19] https://www.unsloth.ai/blog/llama3

[20] https://huggingface.co/docs/diffusers/training/lora

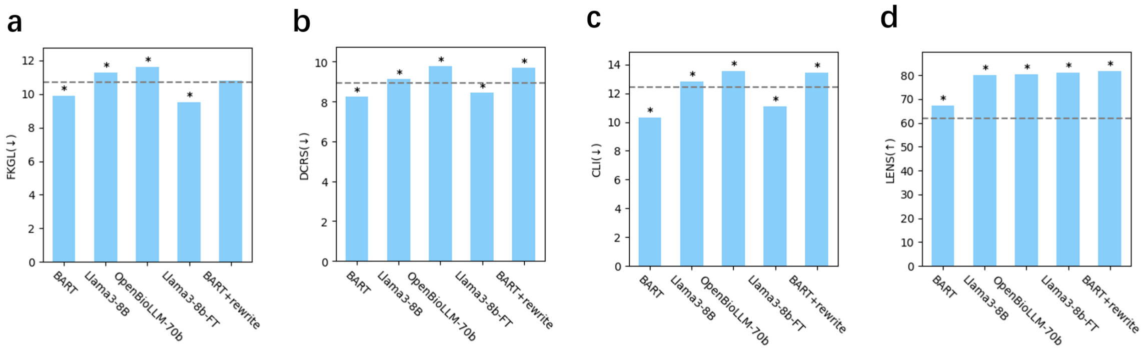


Figure 1: A comparison of readability performance of lay summaries generated by different methods on the eLife validation dataset. (a) Median FKGL score, (b) Median DCRS score, (c) Median CLI score, (d) Median LENS score. In each figure, the gray horizontal dashed line represents the median readability score of the reference lay summary in the validation set. The (*) symbol indicates that the Wilcoxon signed-rank test was passed (p-value less than 0.05), meaning that the readability score of the generated lay summary is significantly better or worse than that of the reference lay summary. From (d), it can be observed that leveraging the powerful language expression capabilities of LLM significantly enhances the LENS scores of the generated lay summary.

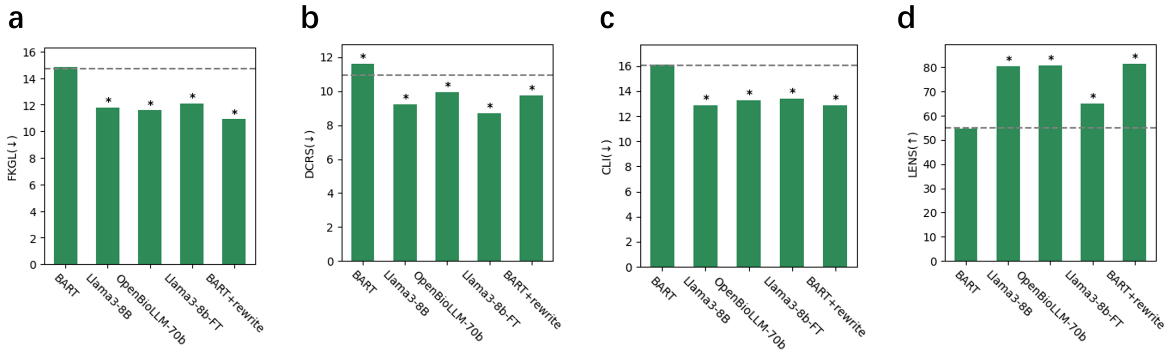


Figure 2: Analysis on models' readability performance on the PLOS validation dataset. (a) Median FKGL score, (b) Median DCRS score, (c) Median CLI score, (d) Median LENS score. gray horizontal dashed line represents the median readability score of the reference lay summary in the validation set. The (*) symbol indicates that the Wilcoxon signed-rank test was passed (p-value less than 0.05), meaning that the readability score of the generated lay summary is significantly better or worse than that of the reference lay summary. We found that the lay summaries written by authors in the PLOS journal generally have significantly lower scores in (a), (b), (c), and (d) readability compared to those generated with the help of LLM.

Eulerian at BioLaySumm: Preprocessing Over Abstract is All You Need

Satyam Modi*
Indian Institute of Technology, Delhi
smodi50448@gmail.com

T Karthikeyan*
Indian Institute of Technology, Delhi
tkarthikeyanai@gmail.com

Abstract

In this paper, we present our approach to the BioLaySumm 2024 Shared Task on Lay Summarization of Biomedical Research Articles at BioNLP workshop 2024 (Goldsack et al., 2024). The task aims to generate lay summaries from the abstract and main texts of biomedical research articles, making them understandable to lay audiences. We used some preprocessing techniques and finetuned Flan-T5 models for the summarization task. Our method achieved an AlignScore of 0.9914 and a SummaC metric score of 0.944. Notably, we scored the highest on the Factuality metric, composed of AlignScore and SummaC, among all the teams.

1 Introduction

Research in every domain has increased significantly, making it challenging for cross-domain researchers to keep track of terminologies outside their expertise. Providing layman summarization in biomedical research addresses this issue. This task is particularly important given the growing volume of biomedical literature, which makes manual summarization impractical. Automated lay summarization can significantly enhance the reach and impact of scientific findings by making them accessible to a wider audience, including patients, healthcare providers, policymakers, and the general public.

The BioLaySumm 2024 Shared Task on Lay Summarization of Biomedical Research Articles is designed to advance the development of automated systems capable of generating accurate and coherent lay summaries from biomedical articles. This task utilizes two separate datasets, focusing on generating summaries that maintain the essence and factuality of the original research while being understandable to a lay audience.

2 Related Work

Past works in summarisation has been along two directions: extractive summarisation and abstractive summarisation. Extractive summarisation involves selecting and extracting key phrases, sentences, or segments directly from the original text to create a summary while in abstractive summarisation the summary is generated by creating new sentences that convey the key information from the original text. Recent works like PEGASUS (Zhang et al., 2020a) uses transformer like models with a self supervised objective for summarisation. In recent years, most of the work on abstractive summarisation has been based on treating the task as a sequence-to-sequence task and using pretrained encoders (Liu and Lapata, 2019).

In this work, we explore on the usage of LLMs for biomedical articles summarisation. Specifically, we use Flan-T5 model (Chung et al., 2022) for finetuning it for our use case by treating summarisation as a sequence-to-sequence task.

3 Datasets

The task included two datasets, PLOS and eLife (Goldsack et al., 2022). PLOS is the larger dataset derived from Public Library of Science, comprising 24,773 instances for training and 1,376 for validation whereas the eLife dataset was derived from the peer-reviewed eLife journal and it contains 4,346 instances for training and 241 for validation. The test data used for evaluation consisted of 142 articles each of PLOS and elife datasets.

4 Methodology

4.1 PoA(Preprocessing over Abstract)

The PoA(Preprocessing over Abstract) involves extracting the initial sentences from the research paper which mainly comprises of the abstract and provide a concise overview of the study. Then we

*These authors contributed equally to this work.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 826–830
August 16, 2024. ©2024 Association for Computational Linguistics

apply a regular expression to remove content with parentheses, braces and brackets. These segments often contain supplementary details that can be omitted for a lay audience. This preprocessing step aims to improve readability without compromising the core information.

4.2 Finetuning Flan T5 Models

In our experiments, we fine-tuned various versions of the Flan-T5 model to enhance their performance in summarizing biomedical research articles. Input was the preprocessed abstract obtained from the PoA technique(Section: 4.1) and output was the summary provided in the training data. We began with the Flan-T5 small model, initially fine-tuning it on the PLOS dataset alone.

Next, we expanded the training data to include both PLOS and eLife articles, aiming to improve the model's generalization and robustness. By incorporating a larger and more diverse dataset, we hypothesized that the model would generate more accurate and comprehensive summaries.

We then progressed to fine-tuning the Flan-T5 base model, also using the combined PLOS and eLife datasets. The base model, being larger and more complex than the small model, was expected to capture more intricate patterns and dependencies in the data.

In our final experiment, we applied a cosine scheduler during the fine-tuning of the Flan-T5 base model with the combined datasets. The cosine scheduler adjusts the learning rate dynamically, aiming to improve convergence and model performance by reducing the learning rate gradually, which helps in avoiding overfitting and ensuring better generalization.

5 Experiments and Results

5.1 Hyperparameters for reproducibility

All experiments utilized a batch size of 25, a max input token length of 512, and a max output token length of 300. The learning rate was set to 1e-3. These experiments were conducted on a single NVIDIA A100 40GB GPU for 25 epochs. The code[1] used in this research is publicly accessible.

5.2 Evaluation Metrics

The submission was evaluated across three dimensions: relevance, readability, and factuality. Relevance is measured through metrics including Rouge-1, Rouge-2, Rouge-L (Lin, 2004) and BERTScore (Zhang et al., 2020b). Readability is assessed via the Flesch-Kincaid Grade Level (FKGL) (Kincaid et al., 1975), CLI (Coleman Liau Index) , Dale-Chall Readability Score (DCRS) (Dale and Chall, 1948) and LENS (Maddela et al., 2023). Factuality is measured utilizing AlignScore (Zha et al., 2023) and SummaC (Laban et al., 2021). The scores calculated for each metric are the average of those calculated independently for the generated lay summaries of PLOS and eLife. The aim is to have higher relevance and factuality scores. All the readability scores must be low except the LENS metric.

5.3 Main Results

The evaluation of various Flan-T5 models and the PoA technique yielded several notable observations, which are summarized below:

5.3.1 Flan-T5 Small: PLOS vs. PLOS + eLife Data

When comparing the Flan-T5 small model trained on PLOS data alone to the same model trained on combined PLOS and eLife data, it was observed that the latter configuration was beneficial across all ROUGE scores and readability metrics, indicating better performance in capturing relevant content and readability. However, this enhancement comes at the cost of factuality metrics, as demonstrated by a decrease in AlignScore and SummaC values.

5.3.2 Flan-T5 Small vs. Flan-T5 Base: Combined Data

Comparing the Flan-T5 small and Flan-T5 base models, both trained on the combined PLOS and eLife datasets, revealed that the base model exhibited superior performance in almost all the relevance and readability metrics, with the exception of the DCRS metric, which did not show improvement. Despite these gains, the factuality metrics (AlignScore and SummaC) were compromised in the Flan-T5 base model compared to the small model.

[1] Available at `https://github.com/tkarthikeyan132/PoA`

		Relevance				Readability				Factuality	
Model	**Training data**	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**	**BERTScore**	**FKGL**	**DCRS**	**CLI**	**LENS**	**AlignScore**	**SummaC**
PoA	N/A	**0.4302**	**0.1327**	**0.3965**	**0.8571**	15.5542	11.1486	17.2919	37.4521	**0.9914**	**0.944**
Flan-T5 small	PLOS	0.3935	0.1152	0.3589	0.8479	14.832	11.3634	16.8313	48.7148	0.9369	0.8732
Flan-T5 small	PLOS + eLife	0.4035	0.1166	0.371	0.8451	14.7954	10.7561	16.5336	48.4619	0.9173	0.8538
Flan-T5 base	PLOS + eLife	0.4228	0.1255	0.3879	0.8511	**14.2915**	10.7817	**16.1177**	52.1659	0.8858	0.8024
Flan-T5 base	PLOS + eLife	0.4277	0.1297	0.3942	0.8501	15.0451	**10.6537**	16.6125	**52.3009**	0.9122	0.8385

Table 1: Inference Results of Flan-T5 Models

5.3.3 Flan-T5 Base: With vs. Without Cosine Scheduler

When analyzing the impact of incorporating a cosine learning rate scheduler in the training of the Flan-T5 base model with combined data, it was evident that the scheduler contributed to better readability and factuality metrics. Improvements were noted in DCRS and LENS, while FKGL and CLI metrics became little worse, which are also indicators of readability, were slightly compromised. This suggests that the scheduler helps in fine-tuning the model to better balance readability and factual accuracy.

5.3.4 PoA Technique Performance

Interestingly, the PoA (Preprocessing over Abstract) technique, which does not involve any training, outperformed all Flan-T5 models in terms of relevance and factuality metrics. This technique's performance in ROUGE scores and factuality assessments (AlignScore and SummaC) was superior, highlighting its effectiveness in generating concise and accurate summaries directly from the abstracts. However, the readability scores were lower, likely because abstracts are inherently complex and may not be easily readable by a lay audience.

These findings are detailed in Table 1 illustrating the performance metrics across different models and configurations.

6 Conclusion

The comparative analysis of various Flan-T5 models and the PoA technique for summarizing biomedical research articles has yielded insightful findings. The Flan-T5 small model showed enhanced relevance and readability metrics when trained on combined PLOS and eLife datasets, though at the expense of factuality. The Flan-T5 base model further improved relevance and readability metrics but also compromised factuality. Introducing a cosine learning rate scheduler to the Flan-T5 base model improved readability and factuality metrics, indicating a better balance in model performance.

Notably, the PoA technique, despite not involving any training, outperformed all Flan-T5 models in relevance and factuality metrics, demonstrating its effectiveness in generating accurate and concise summaries from abstracts. These results underscore the importance of training strategies in developing effective summarization models, while also highlighting the potential of simple preprocessing techniques like PoA.

7 Future Scope

The future scope of this research includes augmenting the training datasets to encompass a broader range of biomedical text per article, thereby enhancing the model's generalizability across diverse terminologies and styles. Advanced fine-tuning techniques such as mixed precision training and curriculum learning could be explored to further improve performance in relevance, readability, and factuality. Tailoring models for specific sub-domains within biomedical research could improve accuracy and relevance for specialized fields. Moreover, creating comprehensive evaluation frameworks that consider user satisfaction and practical utility alongside traditional metrics will be essential. Addressing these avenues can significantly advance the effectiveness and applicability of summarization models for biomedical research articles.

8 Limitations

Although we experimented with text-to-text models like Flan-T5, extending our research to autoregressive large language models such as LLaMA 3(AI@Meta, 2024) could offer different advantages and improvements in summarization tasks.

Our experiments focused on preprocessing techniques and hyperparameter tuning, but the potential of prompt tuning with advanced models like GPT-4(et al., 2023) and Gemini(Team et al., 2023) remains unexplored. Investigating prompt engineering and tuning could enhance summarization performance.

Additionally, we combined eLife and PLOS datasets to train a single model, which may not capture the nuances of each dataset. Training separate models for each dataset could yield more specialized and effective summarization capabilities.

Furthermore, our proposed technique might be more effective when integrated into a more complex pipeline to refine the generated summaries. Future research should address these areas to enhance the robustness and applicability of summarization models.

9 Acknowledgements

The authors thank IIT Delhi HPC facility for computational resources.

A Experiments with Various Schedulers

We finetuned the Flan-T5 base model with three distinct schedulers: Cosine, Step, and Exponential. The goal was to determine the impact of each scheduler on the model's performance across multiple metrics. In Table 2, Our experiments demonstrate that the choice of scheduler can significantly impact the performance of the Flan-T5 model in terms of relevance, readability, and factuality. The Cosine scheduler performed best overall in relevance metrics, while the Step scheduler excelled in readability, and the Exponential scheduler achieved the highest factuality scores.

B Experiments with Various Learning Rates

In Table 3, We present the results of experiments conducted to evaluate the performance of the Flan-T5 base model with different learning rates. The learning rates tested in these experiments were 1e-3, 1e-4, 5e-4, and 1e-5. The learning rate of 1e-3 generally provided the best balance across relevance and readability metrics, while the learning rate of 1e-5 excelled in factuality.

C Experiments with and without Preprocessing over Abstract (PoA)

In Table 4, the experiments demonstrate that the PoA method has a nuanced impact on the performance of the Flan-T5 base model. While it slightly reduced some relevance metrics, it improved the depth of content coverage and significantly enhanced factual accuracy. The readability metrics presented mixed results, indicating that the preprocessing step altered the text complexity and structure. These findings suggest that while the PoA method can enhance certain aspects of summarization, it may need to be combined with other techniques for optimal performance across all metrics.

References

AI@Meta. 2024. Llama 3 model card.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *Preprint*, arXiv:2210.11416.

Edgar Dale and Jeanne S. Chall. 1948. A formula for predicting readability. *Educational Research Bulletin*, 27(1):11–28.

OpenAI Josh et al. 2023. Gpt-4 technical report.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Peter Kincaid, Robert P. Fishburne, Richard L. Rogers, and Brad S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2021. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *Preprint*, arXiv:2111.09525.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

		Relevance				Readability				Factuality	
Model	**Scheduler**	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**	**BERTScore**	**FKGL**	**DCRS**	**CLI**	**LENS**	**AlignScore**	**SummaC**
Flan-T5 base	Cosine	**0.4277**	**0.1297**	**0.3942**	**0.8501**	15.0451	10.6537	16.6125	**52.3009**	0.9122	0.8385
Flan-T5 base	Step	0.4161	0.1233	0.3815	0.8495	**14.7222**	10.9538	**16.5340**	49.3804	0.9148	0.8417
Flan-T5 base	Exponential	0.3571	0.0914	0.3332	0.8252	15.3144	**8.4444**	16.7020	40.3914	**0.9294**	**0.8509**

Table 2: Inference Results of Flan-T5 Models with Various Schedulers

		Relevance				Readability				Factuality	
Model	**Learning rate**	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**	**BERTScore**	**FKGL**	**DCRS**	**CLI**	**LENS**	**AlignScore**	**SummaC**
Flan-T5 base	1e-3	**0.4277**	**0.1297**	**0.3942**	**0.8501**	15.0451	**10.6537**	16.6125	**52.3009**	0.9122	0.8385
Flan-T5 base	1e-4	0.4099	0.1205	0.3766	0.8474	14.7894	10.9620	16.5964	48.0036	0.9294	0.8642
Flan-T5 base	5e-4	0.4172	0.1231	0.3833	0.8497	**14.5144**	10.8936	**16.3596**	49.7981	0.9052	0.8308
Flan-T5 base	1e-5	0.4114	0.1189	0.3784	0.8458	15.1296	10.8945	16.8266	49.3234	**0.9388**	**0.8787**

Table 3: Inference Results of Flan-T5 Models with Various Learning Rates

	Relevance				Readability				Factuality	
PoA (Preprocessing over Abstract)	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**	**BERTScore**	**FKGL**	**DCRS**	**CLI**	**LENS**	**AlignScore**	**SummaC**
No	**0.4323**	0.1315	**0.3989**	0.8484	**14.8658**	11.1500	**16.5704**	**50.4200**	0.9486	0.9204
Yes	0.4302	**0.1327**	0.3965	**0.8571**	15.5542	**11.1486**	17.2919	37.4521	**0.9914**	**0.944**

Table 4: Comparison of Performance with and without POA Method

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. *Preprint*, arXiv:1908.08345.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. LENS: A learnable evaluation metric for text simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16383–16408, Toronto, Canada. Association for Computational Linguistics.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *Preprint*, arXiv:2305.16739.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020a. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. *Preprint*, arXiv:1912.08777.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

HGP-NLP at BioLaySumm: Leveraging LoRA for Lay Summarization of Biomedical Research Articles using Seq2Seq Transformers

Hemang Malik[*†] **Gaurav Pradeep**[‡†] **Pratinav Seth** [*]
Manipal Institute of Technology
Manipal Academy of Higher Education, Manipal, India
{hemangmalik2486,gauravpradeep2004,seth.pratinav}@gmail.com

Abstract

Lay summarization aims to generate summaries of technical articles for non-experts, enabling easy comprehension for a general audience. The technical language used in research often hinders effective communication of scientific knowledge, making it difficult for non-experts to understand. Automatic lay summarization can enhance access to scientific literature, promoting interdisciplinary knowledge sharing and public understanding. This has become especially important for biomedical articles, given the current global need for clear medical information. Large Language Models (LLMs) (Yao et al., 2024) , with their remarkable language understanding capabilities, are ideal for abstractive summarization, helping to make complex information accessible to the public. This paper details our submissions to the BioLaySumm 2024 Shared Task: Lay Summarization of Biomedical Research Articles (Goldsack et al., 2024). We fine-tune and evaluate sequence-to-sequence models like T5 across various training dataset settings and optimization methods such as LoRA (Hu et al., 2021b) for lay summarization. Our submission achieved the 53rd position overall.

1 Introduction

Scientific research aims to advance knowledge, but a major challenge is the lack of domain-specific knowledge among the public. Technical reports and research articles are often incomprehensible to non-experts, hindering knowledge dissemination. Lay summarization addresses this by generating factual, readable summaries of technical texts for non-experts (Chaturvedi et al., 2020). In the biomedical field, lay summarization is crucial due to its highly technical content involving complex medical terms and detailed research findings. Access to clear medical information is essential for informed health decisions. Misunderstandings or a lack of access can lead to poor decisions, increased anxiety, and a general distrust of medical advice (Guo et al., 2021). Recent advancements in large language models (LLMs) (Yao et al., 2024) and autoregressive LLMs (Chen et al., 2023) like ChatGPT, Gemini, Mistral, and Llama have shown significant potential. With their large-scale pretraining, these models can generate high-quality, contextually relevant summaries that are informative and accessible. However, they are computationally expensive and often require fine-tuning for specific tasks, needing substantial computational resources and energy. Sequence-to-sequence models (Chiu et al., 2018) provide a promising alternative, addressing these computational challenges. With an encoder-decoder structure, these models efficiently handle input and output sequences, making them well-suited for summarization (Kouris et al., 2021). They are more computationally efficient than LLMs, requiring fewer resources while delivering high performance. This paper describes our approach to the BioLaySumm 2024 Shared Task: Lay Summarization of Biomedical Research Articles (Goldsack et al., 2024). We fine-tune and evaluate sequence-to-sequence models like T5 for the task of lay summarization (Challagundla and Peddavenkatagari, 2024) across various training dataset settings and optimization methods such as LoRA (Hu et al., 2021b). Our final submission to this task is a LoRA-based Flan-T5-Base model (Rusnachenko and Liang, 2024), which secured us the 53rd position on the leaderboard.

2 Background and Previous Works

2.1 Problem and Data Description

The BioLaySumm shared task at the BioNLP Workshop of ACL 2024(Goldsack et al., 2024) focuses on the abstractive summarization of biomedical ar-

[*] Dept. of Data Science & Computer Applications
[†] Equal Contribution
[‡] Dept. of Computer Science & Engineering

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 831–836
August 16, 2024. ©2024 Association for Computational Linguistics

Attributes	Values
Article	In the USA , more deaths happen...
Headings	Abstract, Introduction...
Keywords	epidemiology and global health...
Id	elife-35500-v1...

Table 1: Text Samples from eLife Training dataset with their corresponding attributes

Dataset	Training	Testing	Validation
PLOS	24,773	142	1,376
eLife	4,346	142	341

Table 2: Frequency of Task labels in dataset

ticles. The goal is to generate summaries that are accessible to non-expert audiences by including more background information and using less technical terminology, effectively creating a "lay summary." Task Definition: Given an article's abstract and main text, participants are required to train a model (or models) to generate a lay summary. Two separate datasets, derived from the biomedical journals PLOS and eLife, are provided for model training and will be used for evaluation. For the final evaluation, submissions will be ranked based on their average performance across both datasets.

BioLaySumm offers two datasets from well-respected scientific journals: the Public Library of Science (PLOS) and eLife. Each dataset provides pairs of original research articles along with their corresponding summaries, written by scientists specifically for a public audience, as indicated in Table 1. The PLOS dataset is the larger of the two, containing 24,773 instances for training, 142 for testing, and 1,376 for validation. The eLife dataset contains 4,346 instances for training, 142 for testing, and 241 for validation, as shown in the Table 2. Each dataset includes train, validation, and test sets in the form of JSONL files. The eLife summaries have an average length of around 300-350 words, whereas the PLOS summaries are shorter, averaging around 160-200 words (Goldsack et al., 2022).

2.2 Automatic Text Summarization

Summarization involves condensing large amounts of text into key points. Extractive summarization, like underlining key passages in a textbook, extracts important sentences directly from the original text. This method is simpler and ensures factual accuracy, but the summaries may lack coherence (Neto et al., 2002). On the other hand, Abstractive summarization aims to understand and rephrase the main ideas in a new, concise way, offering more readable summaries but potentially introducing errors if the model misinterprets the original text (Gupta and Gupta, 2019). It goes beyond copying sentences, striving to grasp the core message of a text. Traditional summarization approaches typically fall into two categories: extractive methods that select and combine important sentences from the source text, and abstractive methods that generate entirely new text that captures the main ideas. This field has seen significant progress with transformer-based methods excelling at understanding complex relationships in text. Recently, large language models (LLMs) (Alberts et al., 2023) utilizing autoregressive techniques have emerged as a powerful approach, offering the ability to not only summarize but also generate creative text formats (Poornash et al., 2023).

2.3 Biomedical Lay Summarization

2.3.1 Biomedical Summary Corpora

In the field of deep learning, large collections of text, known as corpora (Stubbs, 2004), are crucial for training language models. These datasets help machines learn the patterns, meaning, and structure of language, thereby improving their ability to understand and generate text for tasks such as sentiment analysis or machine translation (Gilquin and Gries, 2009). Instead of creating a new corpus from scratch, focus on utilizing existing collections of scientific articles alongside their corresponding, simplified summaries written for the general public. The BioLaySumm datasets (PLOS and eLife) are built by extracting these paired texts from the corpora. These datasets then become the foundation for training and evaluating models that can automatically generate clear summaries of scientific research articles for a non-scientific audience (Goldsack et al., 2022).

2.3.2 Sequence-to-sequence models

Significant research has utilized sequence-to-sequence (seq2seq) models (Sriram et al., 2017) for generating lay summaries of biomedical research articles. Leveraging pre-trained models like T5 and BART (Colak and Karadeniz, 2023), researchers have explored methods to effectively capture the complexities of biomedical language and translate it into clear summaries for the public. Another area of focus involves adapting Transformers (Dandan et al., 2023) by incorporating biomedical domain knowledge, such as using pre-trained models on scientific text or knowledge graphs, to enhance the model's understanding of scientific concepts. Additionally, research (Fan et al., 2018) investigates controllable summarization with seq2seq models, allowing researchers to tailor summaries to specific needs like factual accuracy or readability (Sriram et al., 2017).

2.3.3 Generative Pre-trained Transformers

Generative Pre-trained Transformers (GPT) (Zhu and Luo, 2022) are advanced auto-regressive transformer models trained on vast amounts of data, enabling them to generate human-like text and perform various natural language understanding tasks with high accuracy and efficiency (Luo et al., 2022). Their strong performance is attributed to the quality of the training dataset and their autoregressive nature, where the input serves as the "prefix" of the output. However, to maximize their performance, fine-tuning on specific downstream tasks is necessary, which can be computationally expensive.

2.3.4 Infusing External Knowledge

Integrating external knowledge into models (Paulheim, 2017) involves incorporating domain-specific data or structured information, which boosts performance and contextual comprehension for more accurate predictions and insights. This approach enriches deep learning models (Menghani, 2023) with additional context and expertise, enhancing their robustness and interpretability (Koncel-Kedziorski et al., 2022). One widely used technique involves integrating external knowledge graphs (Paulheim, 2017) for downstream tasks like summarization. This leverages semantic relationships to better understand and contextualize concepts within summaries (Goldsack et al., 2023).

2.4 Current State of NLP

State-of-the-art foundation large language models (LLMs) like ChatGPT (Firat, 2023), Gemini (Khurdula et al., 2024), and LLaMA (Touvron et al., 2023) are revolutionizing how we interact with text. These LLMs are auto-regressive in nature and can generate various creative text formats and translate languages with impressive fluency. However, they still require fine-tuning to achieve state-of-the-art performance in downstream tasks. This can be done by optimizing on a downstream dataset and task or by using methods involving prompting. Fine-tuning these models is computationally expensive due to their large number of parameters. Techniques such as Low-Rank Adaptation of Large Language Models (LoRA) (Hu et al., 2021b) and Parameter-Efficient Fine-Tuning (PEFT) (Sabry and Belz, 2023) address this challenge. These methods enable fine-tuning by freezing the pre-trained model weights and injecting trainable rank decomposition matrices into each layer of the Transformer architecture, greatly reducing the number of trainable parameters for downstream tasks.

3 System Overview

Our approach to the task involved fine-tuning sequence-to-sequence transformer-based models (Ramachandran et al., 2016), such as T5, across various training dataset settings and optimization methods like LoRA on the PLoS and eLife datasets. The models were evaluated based on the readability (Paasche-Orlow et al., 2003), relevance (Montague and Aslam, 2001), and factuality (Pagnoni et al., 2021) of the summaries generated. The model demonstrating the best performance across these three evaluation criteria was chosen as our final submission for the task.

3.1 Data Pre-Processing

Each sample in the dataset has attributes as illustrated in Table 1. These were utilized to design a prompt structured as follows:

> Provide a lay summary of the following article, which includes keywords <keywords>: <article>.

For processing text, the T5 tokenizer was used with a maximum input of 1024 tokens. LoRA-based models, on the other hand, could handle up to 2048 tokens as input (Seye et al., 2018). In all cases, the output was capped at 512 token.

3.2 Fine-Tuning Transformers

T5 is a text-to-text Transformer architecture that treats all tasks, like translation and classification, as generating target text from input text. It differs from BERT by adding a causal decoder and using various pre-training tasks instead of the cloze task. We trained the sequence-to-sequence transformer T5 on the given datasets using different approaches. This included training solely on either the eLife or PLoS dataset, followed by inference on the evaluation and test sets. Alternatively, we also trained

Model	Data	Relevance				Readability				Factuality	
Name	Type	R-1	R-2	R-L	BTs	FL	DS	CI	LS	As	Sc
T5-small	1	**0.297**	0.086	**0.269**	0.837	11.201	9.913	12.786	44.223	0.794	0.741
	2	0.296	0.087	0.268	**0.838**	**11.351**	**10.158**	**13.135**	**48.178**	**0.840**	**0.766**
LORA Flan T5-base	1	0.296	**0.087**	0.268	**0.838**	**11.351**	**10.158**	**13.135**	**48.178**	**0.840**	**0.766**

Table 3: Results of Experimented Models on the Test Set.'Data type' column indicates the number of models used: -1 (unified) or 2 (one for each dataset). Here, R=ROUGE F1, BTs=BERTScore, FL=Flesch-Kincaid Grade Level, DS=Dale-Chall Readability Score, CI=Coleman-Liau Index, LS=LENS metric, As=AlignScore, and Sc=SummaC.

the model on a combined dataset consisting of both train sets before performing inference. Flan-T5 is an extension of T5 (Rusnachenko and Liang, 2024) that has been further fine-tuned on a diverse set of instructions, enhancing its performance on various downstream tasks, including summarization. To fine-tune this model, we employed the Parameter Efficient Fine Tuning Method, specifically the Low-Rank Adaptation (LoRA) technique (Hu et al., 2021a), which efficiently fine-tunes pre-trained models by adapting a small number of model parameters. This method significantly reduces computational requirements while improving performance.

3.3 Implementation Details

The experiments were conducted with a learning rate of 1e-3. For fine-tuning the T5-base (Guan et al., 2024) using LORA (Hu et al., 2021a), we selected a rank of 32, a LoRA alpha value of 32, and a LoRA dropout of 0.05. The T-5 small model was trained for 10 epochs while the Flan-T5 model was trained for 2 epochs. All experiments were carried out on a P100 GPU via Kaggle.

4 Analysis & Results

The challenge provides two individual datasets of different distributions for training, evaluation, and testing. Details about the distribution of the datasets are illustrated in Table 2. To evaluate the lay summary, we have taken into account three major factors: readability, relevance, and factuality. Readability scores, i.e., Flesch-Kincaid Grade Level (FKGL), Dale-Chall Readability Score (DCRS), Coleman-Liau Index (CLI), and LENS assess how easy summaries are to understand for a lay audience. Relevance scores i.e., ROUGE (1, 2, and L) and BERTScore measure how well summaries capture key points from the original research. Factuality scores i.e., AlignScore, SummaC evaluate how accurately summaries reflect factual information.

The detailed results of our experiments are presented in **Table 3**. Although none of the models achieved high scores, this can be attributed to the lack of constraints and the limited training and context length of the models. This leads to all models converging to similar scores despite differences in model sizes and training methodologies. However, it is evident that training on both datasets together improves the readability and factuality of the models. This improvement is also observed when fine-tuning a large model using parameter-efficient fine-tuning techniques like LORA (Hu et al., 2021b) , which is originally pre-trained on a large text corpus. This might be due to the fact that PEFT (Pu et al., 2023) techniques enable large language models to learn more effectively while retaining previously acquired knowledge. However, this claim for lay summarization would require further experiments to validate.

5 Conclusion

We fine-tune and evaluate sequence-to-sequence models like T5 for the task of lay summarization (Challagundla and Peddavenkatagari, 2024) across various training dataset settings and optimization methods such as LoRA. We extended our evaluation beyond traditional accuracy metrics to encompass real-world application considerations like relevance (Montague and Aslam, 2001) , readability (Paasche-Orlow et al., 2003) , and factuality (Pagnoni et al., 2021) . Our analysis revealed interesting trade-offs of fine-tuning pre-trained models using traditional and parameter-efficient methods like LoRA (Hu et al., 2021b). Future work should explore the impact of PEFT-similar models for lay and abstractive summarization.

Limitations

Due to computational resource limitations, we were able to conduct only a limited number of experiments and were constrained by input token limits. Our access was limited to the Kaggle P100 GPU, for which we are grateful. This restriction led us to primarily experiment with smaller model sizes, potentially missing the benefits of larger architectures. Our hyperparameter tuning was not extensive, and as a result, our models' performance fell short of high scores on evaluation metrics. This indicates substantial room for improvement. We focused narrowly on sequence-to-sequence models, and these limitations present clear opportunities for future research to build upon and investigate.

Acknowledgments

We would like to thank the Research Society MIT, an undergraduate interdisciplinary technical society of Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India, for supporting our research.

References

Ian L Alberts, Lorenzo Mercolli, Thomas Pyka, George Prenosil, Kuangyu Shi, Axel Rominger, and Ali Afshar-Oromieh. 2023. Large language models (llm) and chatgpt: what will the impact on nuclear medicine be? *European journal of nuclear medicine and molecular imaging*, 50(6):1549–1552.

Bhavith Chandra Challagundla and Chakradhar Peddavenkatagari. 2024. Neural sequence-to-sequence modeling with attention by leveraging deep learning architectures for enhanced contextual understanding in abstractive text summarization. *arXiv preprint arXiv:2404.08685*.

Rochana Chaturvedi, Jaspreet Singh Dhani, Anurag Joshi, Ankush Khanna, Neha Tomar, Swagata Duari, Alka Khurana, Vasudha Bhatnagar, et al. 2020. Divide and conquer: From complexity to simplicity for lay summarization. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 344–355.

Yongchao Chen, Jacob Arkin, Yang Zhang, Nicholas Roy, and Chuchu Fan. 2023. Autotamp: Autoregressive task and motion planning with llms as translators and checkers. *arXiv preprint arXiv:2306.06531*.

Chung-Cheng Chiu, Tara N Sainath, Yonghui Wu, Rohit Prabhavalkar, Patrick Nguyen, Zhifeng Chen, Anjuli Kannan, Ron J Weiss, Kanishka Rao, Ekaterina Gonina, et al. 2018. State-of-the-art speech recognition with sequence-to-sequence models. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 4774–4778. IEEE.

Cagla Colak and Lknur Karadeniz. 2023. ISIKSumm at BioLaySumm task 1: BART-based summarization system enhanced with bio-entity labels. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 636–640, Toronto, Canada. Association for Computational Linguistics.

Wang Dandan, Hong He, and Cong Wei. 2023. Wang et al. 2023. cellular and potential molecular mechanisms underlying transovarial transmission.

Angela Fan, David Grangier, and Michael Auli. 2018. Controllable abstractive summarization. pages 45–54.

Mehmet Firat. 2023. How chat gpt can transform autodidactic experiences and open education?

Gaëtanelle Gilquin and Stefan Th. Gries. 2009. Corpora and experimental methods: A state-of-the-art review. *Corpus Linguistics and Linguistic Theory*, 5(1):1–26.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chen Tang, Carolina Scarton, and Chenghua Lin. 2023. Enhancing biomedical lay summarisation with external knowledge graphs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8016–8032, Singapore. Association for Computational Linguistics.

Boxu Guan, Xinhua Zhu, and Shangbo Yuan. 2024. A t5-based interpretable reading comprehension model with more accurate evidence training. *Information Processing & Management*, 61(2):103584.

Yue Guo, Wei Qiu, Yizhong Wang, and Trevor Cohen. 2021. Automated lay language summarization of biomedical scientific reviews. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 160–168.

Som Gupta and S. K Gupta. 2019. Abstractive summarization: An overview of the state of the art. *Expert Systems with Applications*, 121:49–65.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021a. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021b. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685.

Harsha Vardhan Khurdula, Anbilparithi Pagutharivu, and Jin Soung Yoo. 2024. The future of feelings: Leveraging bi-lstm, bert with attention, palm v2 & gemini pro for advanced text-based emotion detection. In *SoutheastCon 2024*, pages 275–278. IEEE.

Rik Koncel-Kedziorski, Dhanush Bekal, Yi Luan, Mirella Lapata, and Hannaneh Hajishirzi. 2022. Text generation from knowledge graphs with graph transformers. *Preprint*, arXiv:1904.02342.

Panagiotis Kouris, Georgios Alexandridis, and Andreas Stafylopatis. 2021. Abstractive text summarization: Enhancing sequence-to-sequence models using word sense disambiguation and semantic content generalization. *Computational Linguistics*, 47(4):813–859.

Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. 2022. BioGPT: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6):bbac409.

Gaurav Menghani. 2023. Efficient deep learning: A survey on making deep learning models smaller, faster, and better. *ACM Computing Surveys*, 55(12):1–37.

Mark Montague and Javed A Aslam. 2001. Relevance score normalization for metasearch. In *Proceedings of the tenth international conference on Information and knowledge management*, pages 427–433.

Joel Larocca Neto, Alex A. Freitas, and Celso A. A. Kaestner. 2002. Automatic text summarization using a machine learning approach. In *Advances in Artificial Intelligence*, pages 205–215, Berlin, Heidelberg. Springer Berlin Heidelberg.

Michael K Paasche-Orlow, Holly A Taylor, and Frederick L Brancati. 2003. Readability standards for informed-consent forms as compared with actual readability. *New England journal of medicine*, 348(8):721–726.

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. 2021. Understanding factuality in abstractive summarization with frank: A benchmark for factuality metrics. *arXiv preprint arXiv:2104.13346*.

Heiko Paulheim. 2017. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web*, 8(3):489–508.

A.s. Poornash, Atharva Deshmukh, Archit Sharma, and Sriparna Saha. 2023. APTSumm at BioLaySumm task 1: Biomedical breakdown, improving readability by relevancy based selection. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 579–585, Toronto, Canada. Association for Computational Linguistics.

George Pu, Anirudh Jain, Jihan Yin, and Russell Kaplan. 2023. Empirical analysis of the strengths and weaknesses of peft techniques for llms. *arXiv preprint arXiv:2304.14999*.

Prajit Ramachandran, Peter J Liu, and Quoc V Le. 2016. Unsupervised pretraining for sequence to sequence learning. *arXiv preprint arXiv:1611.02683*.

Nicolay Rusnachenko and Huizhi Liang. 2024. nicolay-r at semeval-2024 task 3: Using flan-t5 for reasoning emotion cause in conversations with chain-of-thought on emotion states. *arXiv preprint arXiv:2404.03361*.

Mohammed Sabry and Anya Belz. 2023. Peft-ref: A modular reference architecture and typology for parameter-efficient finetuning techniques. *arXiv preprint arXiv:2304.12410*.

Madoune R Seye, Bassirou Ngom, Bamba Gueye, and Moussa Diallo. 2018. A study of lora coverage: Range evaluation and channel attenuation model. In *2018 1st International Conference on Smart Cities and Communities (SCCIC)*, pages 1–4. IEEE.

Anuroop Sriram, Heewoo Jun, Sanjeev Satheesh, and Adam Coates. 2017. Cold fusion: Training seq2seq models together with language models. *arXiv preprint arXiv:1708.06426*.

Michael Stubbs. 2004. Language corpora. *The handbook of applied linguistics*, pages 106–132.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Yifan Yao, Jinhao Duan, Kaidi Xu, Yuanfang Cai, Zhibo Sun, and Yue Zhang. 2024. A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. *High-Confidence Computing*, page 100211.

Qihao Zhu and Jianxi Luo. 2022. Generative pre-trained transformer for design concept generation: an exploration. *Proceedings of the design society*, 2:1825–1834.

Ctyun AI at BioLaySumm: Enhancing Lay Summaries of Biomedical Articles Through Large Language Models and Data Augmentation

Ruijing Zhao[1,†,*], Siyu Bao[1,†,*], Siqin Zhang[1], Jinghui Zhang[1], Weiyin Wang[1], Yunian Ru[1],
[1]China Telecom Cloud Technology Co., Ltd
{zhaorj1,baosy,zhangsq20,zhangjh33,wangwy23,ruyn}@chinatelecom.cn
[†] Corresponding author

Abstract

Lay summaries play a crucial role in making scientific research accessible to a wider audience. However, generating lay summaries from lengthy articles poses significant challenges. We consider two approaches to address this issue: Hard Truncation, which preserves the most informative initial portion of the article, and Text Chunking, which segments articles into smaller, manageable chunks. Our workflow encompasses data preprocessing, augmentation, prompt engineering, and fine-tuning large language models. We explore the influence of pretrained model selection, inference prompt design, and hyperparameter tuning on summarization performance. Our methods demonstrate effectiveness in generating high-quality, informative lay summaries, achieving the second-best performance in the BioLaySumm shared task at BioNLP 2024.

1 Introduction

Biomedical publications serve as a critical channel for disseminating cutting-edge research findings on a wide range of health-related topics. While biomedical publications are essential for advancing medical knowledge and public health awareness, the technical terminology and lack of backgroud information often render them inaccessible to non-expert audiences(Guo et al., 2021). The BioLaySumm shared task addresses this need by developing effective models to generate lay summaries of biomedical articles aimed at non-expert audiences(Goldsack et al., 2024).

The challenge in the BioLaySumm shared task is to distill complex biomedical content into lay summaries that are both comprehensible and engaging to non-expert audiences. Large language models (LLMs) have shown remarkable capabilities in generating coherent and contextually accurate texts(Naveed et al., 2023), which could reformulate complex technical information into simpler narratives(Turbitt et al., 2023). Thus, LLMs are ideal for the generation of lay summaries. LLMs have witnessed the great advancement, each showcasing unique capabilities and specialized applications(Zhao et al., 2023), such as Mistral(Jiang et al., 2023), Qwen(Bai et al., 2023) and Llama(Touvron et al., 2023).

File	Key	Min	Max	Mean	Median
eLife	lay summary	225	893	478	473
	article	444	54,539	16,555	15,866
PLOS	lay summary	17	674	268	270
	article	1,046	37,770	10,289	10,029

Table 1: Token length statistics for the eLife and PLOS datasets, obtained using the Mistral tokenizer.

To tackle the challenge of lengthy articles in the BioLaySumm shared task, we consider two approaches: Hard Truncation and Text Chunking. We preprocess the data using these methods, apply data augmentation and prompt engineering, and fine-tune large language models on the task-specific data. We explore the effect of pretrained models, inference prompts, and hyperparameters on the quality of the generated lay summaries. Our experiments show that our approach effectively extracts key information and produces informative, easy-to-understand summaries.

2 Related Work

2.1 Large Languange Model Generation

Recent advancements in generation models have been dominated by the emergence of LLMs such as Mistral(Jiang et al., 2023), Llama(Touvron et al., 2023) and GPT-4(OpenAI et al., 2024). In the domain of biomedical summarization, LLMs have been adapted to interpret and summarize complex scientific texts, providing a foundation for tasks like BioLaySumm (Brown et al., 2020). Moreover, text chunking, an essential natural language pro-

[*] These authors contributed equally to this work.

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 837–844
August 16, 2024. ©2024 Association for Computational Linguistics

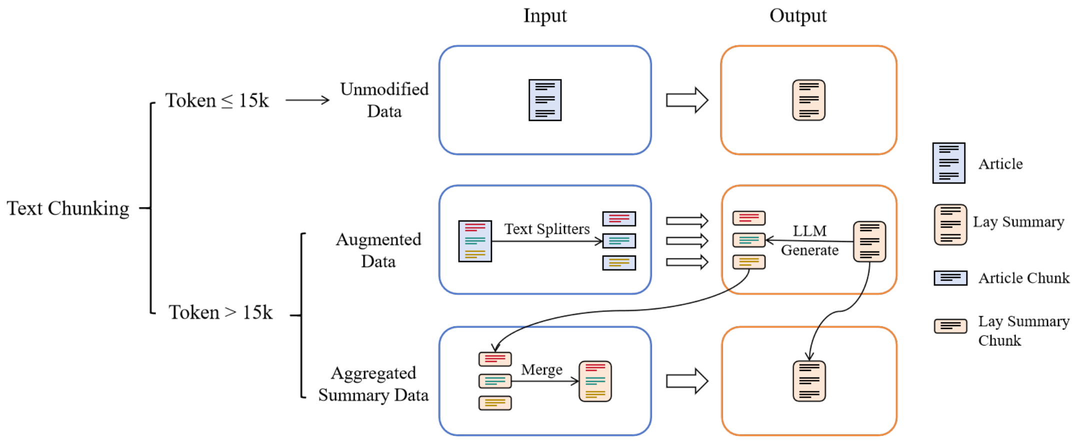


Figure 1: Text Chunking processes articles based on their token count. For articles with fewer than 15k tokens, the original content is preserved. Articles exceeding 15k tokens are divided into chunks, and the lay summary is generated using an LLM for each chunk. The generated lay summary chunks are then merged and used as input, with the original lay summary serving as the output.

cessing (NLP) technique, plays a critical role in BioLaySumm by breaking down large texts into manageable chunks(Reddy et al., 2023). This process enhances the accuracy of embedded content and improves important information retrieval, thereby enhancing the efficiency and quality of text retrieval and generation in the biomedical field.

2.2 Data Augmentation

Data augmentation (Shorten et al., 2021) in LLMs involves enriching the training dataset with artificially generated samples, which enhances the model's robustness and generalization capabilities. In biomedical summarization, data augmentation techniques such as back-translation (Sugiyama and Yoshinaga, 2019) and paraphrasing(Mi et al., 2022) have been used to expand the diversity of training examples, helping models to better handle a range of linguistic structures and terminologies found in medical texts (Li et al., 2022).

3 Data Preprocessing

3.1 Dataset

The dataset for BioLaySumm shared task is a combination of two biomedical datasets, PLOS and eLife(Goldsack et al., 2022). These datasets contain research articles and corresponding lay summaries written by experts .The diversity of these datasets presents a challenge for participants in developing models that effectively summarize biomedical literature for a general audience.

Between the two provided datasets, PLOS is larger, with 24,773 instances for training and 1,376 for validation, while eLife has 4,346 training instances and 241 validation instances.

3.2 Optimizing Input Article

Given the computational constraints, we limit the maximum context length to 15k tokens. Table 1 presents the token length statistics in the eLife and PLOS datasets. The statistics reveal that a considerable number of articles surpass the 15k token limit. We evaluate two approaches to address this challenge when applying Supervised Fine-Tuning (SFT) to adapt pretrained language models for specific tasks: Hard Truncation and Text Chunking.

Hard Truncation: This approach truncates articles, keeping only the first 15k tokens. It relies on the typical structure of articles, where crucial information is often presented initially. Truncating the latter part minimizes the loss of critical information while using only the provided data corpus. However, for longer articles, it may lead to information loss and potentially cause the model to generate content not present in the input.

Text Chunking: As shown in Figure 1, Text Chunking uses Langchain's Text Splitters* to divide articles into chunks of 15k tokens or less. This ensures the entire article is used in the SFT data. However, chunking introduces artificial boundaries within the text, which may disrupt the natural flow and context of the article, potentially impacting model performance. It also increases the number of training data entries, as a single entry may be

*https://python.langchain.com/v0.1/docs/modules/data_connection/document_transformers/recursive_text_splitter/

split into multiple chunks. This could result in longer articles having a disproportionate influence on the training process, as they contribute more chunks to the dataset.

We evaluate both methods on different datasets to determine the most optimal approach for each.

3.3 Data Augmentation

Hard Truncation does not introduce new content, but Text Chunking splits articles into fragments that do not match the original lay summaries. To address this issue, we use data augmentation with Mixtral 8x7B (Jiang et al., 2024) (hereafter Mixtral). Mixtral generates lay summaries for these fragments by finding the corresponding content from the full-text lay summary. It uses the original text as much as possible.

To include the full-text lay summary in the training data, we use the Mixtral-generated summaries as input and the original full-text summary as output. This incorporates the full-text summary into the training process for Text Chunking.

Data augmentation with Mixtral generates summaries that accurately correspond to the article fragments from Text Chunking. It also ensures the full-text summary is included in the training data.

3.4 Prompt Engineering for Data Segregation

For the Hard Truncation approach, a uniform prompt is used for all data entries. However, the Text Chunking method requires different prompts for three data types:

Unmodified Data: Articles not exceeding 15k tokens are retained directly and form the main portion of the training data. The prompt used for this data type is consistent with the one used during inference.

Augmented Data from Chunking: For articles split into chunks, the input text consists of the article chunk, while the output text is generated using Mixtral. A different prompt is employed during training to differentiate it from unmodified data.

Aggregated Summary Data: The outputs from augmented data from chunking are concatenated in the article's narrative order. This concatenated text serves as the input, and the original lay summary is used as the output. The prompt instructs the model to generate a concise lay summary from the overly long and redundant input.

The specific prompts used for each data type are presented in Table 6 of the Appendix.

4 Metrics

To thoroughly evaluate the quality of the generated lay summaries, we use a diverse set of metrics that capture various aspects of the summarization task:

Relevance: We use ROUGE (1, 2, and L) (Lin, 2004) and BERTScore (Zhang et al., 2019) to evaluate the relevance of the generated summaries to the original articles. Higher scores indicate better performance for these metrics.

Readability: To assess the readability of the generated summaries, we utilize several widely-used metrics: Flesch-Kincaid Grade Level (FKGL) (Kincaid et al., 1975), Dale-Chall Readability Score (DCRS) (Chall and Dale, 1995), Coleman-Liau Index (CLI) (Coleman and Liau, 1975), and LENS (Maddela et al., 2022). For FKGL, DCRS, and CLI, lower scores indicate better readability, while for LENS, higher scores are preferable.

Factuality: Ensuring the factual correctness of the generated summaries is crucial in the biomedical domain. We employ AlignScore (Zha et al., 2023) and SummaC (Laban et al., 2022) to measure the factual consistency between the generated summaries and the source articles. Higher scores on these metrics indicate better factual alignment.

5 Experiments

We conduct a series of experiments to investigate the impact of various factors on our lay summarization model's performance. Due to the PLOS validation set's size, we use the first 142 entries as our validation subset.

5.1 Impact of the Pretrained Model

We compare the performance of three pretrained language models: Qwen1.5-14B-Chat, Mistral-7B-Instruct-v0.2, and Meta-Llama-3-8B-Instruct. Each model is fine-tuned on the Hard Truncation dataset for one epoch with a learning rate of 1e-5 and a global batch size of 64. We use a complex prompt during inference, described in Section 5.2.

Table 2 shows the results. Meta-Llama-3-8B-Instruct achieves the highest LENS score but performs worse on other metrics. Qwen1.5-14B-Chat and Mistral-7B-Instruct-v0.2 exhibit comparable performance, with the latter having fewer parameters. Based on these findings, we select Mistral-7B-Instruct-v0.2 as our base model for subsequent experiments.

Model	ROUGE1	ROUGE2	ROUGEL	BERTScore	FKGL ↓	DCRS ↓	CLI ↓	LENS	AlignScore	SummaC
Qwen1.5-14B-Chat	0.4842	0.156	0.454	**0.8677**	**11.537**	9.559	**13.445**	54.865	0.7804	0.6876
Mistral-7B-Instruct-v0.2	**0.4959**	**0.1640**	**0.4654**	0.8672	12.054	**9.4289**	13.5558	52.0932	**0.7954**	**0.7070**
Meta-Llama-3-8B-Instruct	0.473	0.1464	0.4391	0.8581	12.0817	9.8036	13.5764	**66.8112**	0.739	0.6816

Table 2: Experiment results of different pretrained models. For FKGL, DCRS, and CLI, lower scores are better; for all other metrics, higher scores are better.

Prompt	ROUGE1	ROUGE2	ROUGEL	BERTScore	FKGL ↓	DCRS ↓	CLI ↓	LENS	AlignScore	SummaC
Simple Prompt	0.4804	0.1521	0.4514	0.8661	**11.936**	**9.3647**	**13.407**	**54.716**	0.7783	0.6716
Complex Prompt	**0.4959**	**0.1640**	**0.4654**	**0.8672**	12.054	9.4289	13.5558	52.0932	**0.7954**	**0.7070**
One-shot Prompt	0.4755	0.1496	0.4462	0.8652	12.104	9.4766	13.5491	54.232	0.7799	0.6694

Table 3: Experiment results of different inference prompts.

5.2 Impact of Inference Prompts

We investigate the impact of three distinct inference prompts on model performance: a simple prompt, a complex prompt, and a one-shot prompt. The specific prompts are detailed in Table 7.

Experiments using the Mistral-7B-Instruct-v0.2 model (Table 3) show that the complex prompt yields superior results compared to the simple prompt. The complex prompt improves relevance and factuality but slightly decreases readability. Surprisingly, the one-shot prompt underperforms the other prompts, possibly due to the lengthy example reducing content retention for the predicted sample. We use the complex prompt for subsequent experiments.

5.3 Impact of Hyperparameters

In the process of hyperparameter optimization, we drew inspiration from the experimental configurations employed in the Llama2 study. Our investigation focused on two critical hyperparameters: the number of training epochs and the learning rate. Specifically, we conducted a series of fine-tuning experiments using the Mistral-7B-Instruct-v0.2 model. The experimental design was as follows:

1. Single-epoch training with learning rates of 1e-5 and 2e-5.

2. Comparative analysis of single-epoch and dual-epoch training, both utilizing a learning rate of 1e-5.

This systematic approach allowed us to assess the individual and combined effects of epoch count and learning rate on model performance. By benchmarking against the Llama2 configurations, we aimed to leverage established best practices while adapting them to our specific task requirements. The results of these experiments provided valuable insights into the optimal hyperparameter settings for our fine-tuning process, enabling us to strike a balance between model performance and computational efficiency.

5.4 Impact of Data Augmentation

To address the challenge of articles exceeding 15k tokens, we developed and evaluated two distinct methods: Hard Truncation and Text Chunking. Hard Truncation preserves the original lay summary style but risks omitting content from the latter portions of the article. Conversely, Text Chunking ensures comprehensive inclusion of the entire article in the training set, albeit with the potential introduction of noise during data augmentation.

The application of these methods is contingent upon various factors. Hard Truncation may be more appropriate when less critical information is concentrated at the article's end or when sophisticated models for data transformation are unavailable. However, Text Chunking could potentially yield superior results when crucial content is distributed throughout the article.

To empirically assess the impact of these data processing methods, we fine-tuned separate models using datasets prepared with Hard Truncation and Text Chunking. The results, presented in Table 5, reveal that the Hard Truncation-trained model exhibits superior performance on the eLife dataset, while the Text Chunking-trained model demonstrates enhanced efficacy on the PLOS dataset. Leveraging these findings, we implemented an ensemble approach combining both models for our final submission. This strategy proved effective, securing 3rd place in relevance and 2nd place in the overall ranking of the competition.

6 Discussion

This paper introduces two methods for handling long input sequences in the BioLaySumm task and

Epoch	Learning Rate	ROUGE1	ROUGE2	ROUGEL	BERTScore	FKGL ↓	DCRS ↓	CLI ↓	LENS	AlignScore	SummaC
1	1e-5	**0.4959**	**0.1640**	**0.4654**	0.8672	**12.054**	**9.4289**	**13.5558**	52.0932	**0.7954**	**0.7070**
2	1e-5	0.4914	0.1549	0.4596	**0.8675**	12.217	9.576	13.58	**55.166**	0.76	0.6398
1	2e-5	0.4866	0.154	0.4544	0.866	12.551	9.7017	13.8178	52.575	0.7906	0.6587

Table 4: Experiment results of different hyperparameters.

Dataset	DataType	ROUGE1	ROUGE2	ROUGEL	BERTScore	FKGL ↓	DCRS ↓	CLI ↓	LENS	AlignScore	SummaC
eLife	Hard Truncation	**0.5153**	**0.1560**	**0.4904**	**0.8677**	9.9021	8.2115	11.6322	**62.9878**	0.6746	0.5714
	Text Chunking	0.4806	0.1451	0.4589	0.8642	**9.3846**	**7.9235**	**11.0592**	61.2874	**0.6961**	**0.5831**
PLOS	Hard Truncation	**0.4763**	0.1720	**0.4404**	0.8666	**14.2059**	**10.6464**	**15.4795**	**41.1988**	0.9162	0.8426
	Text Chunking	0.4748	**0.177**	0.4400	**0.8680**	14.644	10.77	15.864	40.742	**0.9558**	**0.8747**

Table 5: Experiment results of different data augmentation methods on eLife and PLOS dataset.

investigates the impact of various factors on generating lay summaries. Fine-tuning the Mistral-7B-Instruct-v0.2 model with specific settings yields strong performance.

Hard Truncation and Text Chunking's effectiveness varies depending on the target dataset. Hard Truncation may lose crucial information from later parts of long articles, potentially affecting summary completeness. Text Chunking, while preserving all content, introduces artificial boundaries that could disrupt context and lead to inconsistencies in generated summaries. Additionally, Text Chunking may result in longer articles having disproportionate influence on the training process. We use data augmentation with Mixtral, which generates summaries for text chunks. However, this approach may bias the model towards Mixtral's summarization style and introduce inconsistencies between fragment summaries and full-text summaries.

Future research could explore larger pretrained models and more sophisticated strategies for handling lengthy inputs. Section-specific summarization techniques could also improve performance.

Carefully designing inference prompts and selecting appropriate hyperparameters are crucial when fine-tuning pretrained language models for specific tasks. We hope our work inspires further research and contributes to developing effective tools for making scientific knowledge more accessible.

7 Limitation

In this study, we conducted a comprehensive analysis of various factors influencing model performance, including pre-trained models, hyperparameters, and data processing techniques. Our investigation, however, did not extend to examining the differential impact of distinct article sections on summary generation. This aspect warrants further exploration, as the introduction and conclusion sections often encapsulate the core content of an article and may hold greater significance for summarization, while body sections typically provide more granular details.

Additionally, to enhance the model's proficiency in specialized biological domains, future work could investigate the efficacy of incremental pretraining. This approach may potentially improve the model's ability to elucidate technical terminology in more accessible language, thereby enhancing the overall quality and comprehensibility of generated summaries.

These unexplored avenues present promising directions for future research, aimed at refining and advancing the performance of summarization models in specialized scientific domains, particularly in the field of biology.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Jeanne Sternlicht Chall and Edgar Dale. 1995. Readability revisited: The new dale-chall readability formula.

Meri Coleman and Ta Lin Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283.

Tomas Goldsack, Carolina Scarton, Matthew Shardlow, and Chenghua Lin. 2024. Overview of the biolaysumm 2024 shared task on the lay summarization of biomedical research articles. In *The 23rd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yue Guo, Wei Qiu, Yizhong Wang, and Trevor Cohen. 2021. Automated lay language summarization of biomedical scientific reviews. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 160–168.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Philippe Laban, Tobias Schnabel, Paul N Bennett, and Marti A Hearst. 2022. Summac: Re-visiting nli-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Bohan Li, Yutai Hou, and Wanxiang Che. 2022. Data augmentation approaches in natural language processing: A survey. *Ai Open*, 3:71–90.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2022. Lens: A learnable evaluation metric for text simplification. *arXiv preprint arXiv:2212.09739*.

Chenggang Mi, Lei Xie, and Yanning Zhang. 2022. Improving data augmentation for low resource speech-to-text translation with diverse paraphrasing. *Neural Networks*, 148:194–205.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Nick Barnes, and Ajmal Mian. 2023. A comprehensive overview of large language models. *arXiv preprint arXiv:2307.06435*.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Venkat praneeth Reddy, Pinnapu Reddy Harshavardhan Reddy, Karanam Sai Sumedh, and Raksha Sharma. 2023. IITR at BioLaySumm task 1:lay summarization of BioMedical articles using transformers. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 625–628, Toronto, Canada. Association for Computational Linguistics.

Connor Shorten, Taghi M Khoshgoftaar, and Borko Furht. 2021. Text data augmentation for deep learning. *Journal of big Data*, 8(1):101.

Amane Sugiyama and Naoki Yoshinaga. 2019. Data augmentation using back-translation for context-aware neural machine translation. In *Proceedings of the fourth workshop on discourse in machine translation (DiscoMT 2019)*, pages 35–44.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Oisín Turbitt, Robert Bevan, and Mouhamad Aboshokor. 2023. MDC at BioLaySumm task 1: Evaluating GPT models for biomedical lay summarization. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 611–619, Toronto, Canada. Association for Computational Linguistics.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *arXiv preprint arXiv:2305.16739*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models. *Preprint*, arXiv:2303.18223.

A Prompts

In this sections, we delineate the specific content of the prompts employed in our experimental framework.

Data Type	**Prompt**
Unmodified Data	Generate a 300-400 word abstract for the given biology research article. Include research question, methods, main findings, implications, and conclusions. Use precise scientific terminology, logical structure, and active voice. Ensure clarity and accuracy.Here is the article:{input}. Please give me the clear abstract.
Augmented Data from Chunking	You will be given a section of a scientific article in the field of biology. Your task is to generate a concise and accurate summary of the key points and findings presented in this section. The summary should capture the main ideas, methods, results, and conclusions, while maintaining the scientific context and terminology used in the original text.Here is the article:{input}
Aggregated Summary Data	You will receive a summary of a biology research article generated by an AI model. However, the summary is too long and needs further refinement. Your task is to create a more concise version, focusing on the most critical information. The refined summary should:1. Maintain key findings, conclusions, and scientific context.2. Use precise, domain-specific terminology.3. Follow a logical structure highlighting main points.4. Aiming for 300-400 words.5. Omit unnecessary details while preserving the core message.6. Use clear, concise language for better readability.By adhering to these guidelines, create a highly refined summary that effectively conveys the essence of the original article.Here is the article:{input}

Table 6: Different prompts used for each data type in the experiments.

Prompt Type	**Prompt**
Simple Prompt	Please read the article given and write an easy-to-understand summary.Given article:{input}
Complex Prompt	Generate a 300-400 word abstract for the given biology research article. Include research question, methods, main findings, implications, and conclusions. Use precise scientific terminology, logical structure, and active voice. Ensure clarity and accuracy.Here is the article:{input}. Please give me the clear abstract.
One-Shot Prompt	Generate a 300-400 word abstract for the given biology research article. Include the research question, methods, main findings, implications, and conclusions. Use precise scientific terminology, a logical structure, and active voice. Ensure clarity and accuracy. The abstract should be written in the following format:{example}.Here is the full text of the research article to be summarized:{input}. Please provide a clear and professional abstract based on the article provided. Thank you!

Table 7: Prompt instructing the model to generate a concise lay summary from an overly long and redundant input summary

Author Index