Team Converge at ProbSum 2023: Abstractive Text Summarization of Patient Progress Notes

Gaurav Kolhatkar, Aditya Paranjape, Omkar Gokhale, Dipali Kadam
Pune Institute of Computer Technology, Pune, India
gauravk403@gmail.com, adifeb24@gmail.com,
omkargokhale2001@gmail.com, ddkadam@pict.edu

Abstract

In this paper, we elaborate on our approach for the shared task 1A issued by BioNLP Workshop 2023 titled "Problem List Summarization." With an increase in the digitization of health records, a need arises for quick and precise summarization of large amounts of records. With the help of summarization, medical professionals can sieve through multiple records in a short span of time without overlooking any crucial point. We use abstractive text summarization for this task and experiment with multiple state-of-the-art models like Pegasus, BART, and T5, along with various pre-processing and data augmentation techniques to generate summaries from patients' progress notes. For this task, the metric used was the ROUGE-L score. From our experiments, we conclude that Pegasus is the best-performing model on the dataset, achieving a ROUGE-L F1 score of 0.2744 on the test dataset (3rd rank on the leaderboard).

1 Introduction

Text summarization is the process of shortening a corpus of text into a smaller version while retaining all the crucial information present in the original text. There are two types of text summarizations: abstractive text summarization and extractive text summarization. Abstractive text summarization identifies the critical points of the data and generates a new summary that captures the data's crux. On the other hand, extractive text summarization generates a summary from the words and phrases present within the original data. Text summarization is used in fields like Bio-medicine, Journalism, Finance, etc.

One such field in which there has been a rise in the use of text summarization is Clinical Medicine. Manually reading through medical notes, hospital progress notes, and daily care notes written by doctors and nurses can be monotonous and time-consuming. However, this can be sped up by leveraging automatic text summarization. With the exponential increase in the amount of data that is digitized and readily available, information overload is bound to occur. The automation of summarizing medical notes can help abate the information and cognitive overload faced by medical professionals daily. Text summarization can help these people effectively filter through a plethora of data and focus only on the significant points. There will also be a decrease in the inevitable human errors if the entire process is automated.

Through this paper, we intend to provide a concise summary containing a list of diagnoses and problems for a patient during hospitalization based on the input given in the form of progress notes. In order to increase efficiency and lower diagnostic errors in hospital care, this task intends to stimulate the development of text summarization models for use in diagnostic decision support systems.

2 Related Work

Work on automatic text summarization started in 1958 when Luhn (1958) proposed a method of extracting summaries of scientific literature. This paved the way for extensive research on using Natural Language Processing techniques for text summarization. Other important works from early research include Baxendale (1958)(sentence position and title of the article were used as features for summarizing documents) and Edmundson (1969)(cue words such as "important" or "crucial" were used in addition to title words and sentence location).

Kupiec et al. (1995) introduced machine learning as a method for text summarization in 1995 by training a Naive Bayes classifier to extract a summary from a text. Subsequently, several machine learning approaches have been proposed that use Naive Bayes or Decision tree classifiers, such as Chuang and Yang (2000), which extracted segments of the sentence using special cues, and Neto et al. (2002), which employed a combination of statistical and linguistic features. In recent years, deep learning

The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks, pages 510–515
July 13, 2023 ©2023 Association for Computational Linguistics

methods have been the modus operandi for text summarization, focusing more on generating abstractive summaries. Encoder-decoder RNNs (Nallapati et al., 2016) and LSTM-CNN frameworks (Song et al., 2019) have been shown to perform well on abstractive text summarization tasks.

The advent of transformer architecture in Vaswani et al. (2017) has coincided with a preference for the use of transformer-based methods for summarization. BERT (Zhang et al., 2019), T5 (Raffel et al., 2020), BART (Lewis et al., 2019), and Pegasus (Zhang et al., 2020) have all been used successfully for abstractive summarization.

There has been a growing focus on text summarization in the biomedical domain. Research in the field gathered momentum at the turn of the century, with ten biomedical text summarization studies published between 1999 and 2003 (Afantenos et al., 2005). Most of the early work in this arena used extractive summarization methods, while abstractive summarization has started gaining traction due to the use of highly sophisticated deep-learning models. Shi et al. (2007) proposed BIOSQUASH, a query-based extractive summarization system that uses domain-specific ontologies to rank sentences. Afzal et al. (2020) proposed Biomed-Summarizer, a framework that uses deep neural networks and RNNs to extract meaningful sentences from biomedical text. Approaches that use a combination of extraction and abstraction have also been used (Shing et al. (2021), Adams et al. (2021)). However, Transformer-based models have been shown to perform best for abstractive summarization tasks (Kondadadi et al. (2021), Kieuvongngam et al. (2020), Krishna et al. (2020), Chintagunta et al. (2021)). We have conducted a set of experiments on transformer-based models to create abstractive summaries from patient daily care notes.

3 Dataset Description

The dataset provided for this task (Gao et al., 2023) was sourced from MIMIC-III (Johnson et al., 2016), a publicly available medical dataset. MIMIC-III consists of de-identified EHR data, which is taken from approximately 60,000 hospital ICU admissions at Beth Israel Deaconess Medical Center in Boston, Massachusetts. The data for this task was sourced from the MIMIC-III dataset. The training set has 765 samples, and the test set has 237 samples. The data contains five features: FILE ID, Subjective Sections, Objective Sections, Assessment, and Summary. Assessment input consists of notes taken by a doctor with details about the patient and their diagnosis. Objective sections consist of detailed information regarding a patient's medication dosage and vitals. The subjective sections contain general observations regarding the patient's progress. The summary feature concisely represents all this information and is the ground truth for this task.

4 Methodology

We train the below-mentioned models on the training data. The hyperparameters for each model are mentioned in its respective section. The results obtained on the test data are mentioned in Table 2.

4.1 Pegasus

Pegasus stands for "Pre-training with Extracted Gap sentences for Abstractive Summarization." The Pegasus model divides its input sentences into sub-sequences and feeds them to a series of transformer layers. Leveraging the principles of the attention mechanism, the transformer layers identify the crucial parts of the sentences and generate a new text containing the original document's crux. We finetuned the pretrained "google/pegasus-large" model available on HuggingFace. The model was

trained for 10 epochs with a learning rate of 1e-4 with a linear learning rate scheduler.

4.2 BART

BART stands for "Bidirectional and Auto-Regressive Transformers." To capture semantics and context more effectively, the BART model uses an encoder-decoder architecture. BART is trained as a denoising autoencoder. The pre-training procedure involves corrupting input text using an arbitrary noising function and making the model generate the original text. BART has shown significant improvement on ROUGE score for tasks such as question answering and summarization. We finetuned the pretrained "facebook/bart-large" model available on HuggingFace. The model was trained for 10 epochs with a learning rate of 1e-4 with a linear learning rate scheduler.

4.3 T5

T5 stands for Text-to-Text Transfer Transformer. The T5 framework enables the use of the same model, loss function, and hyperparameters for any

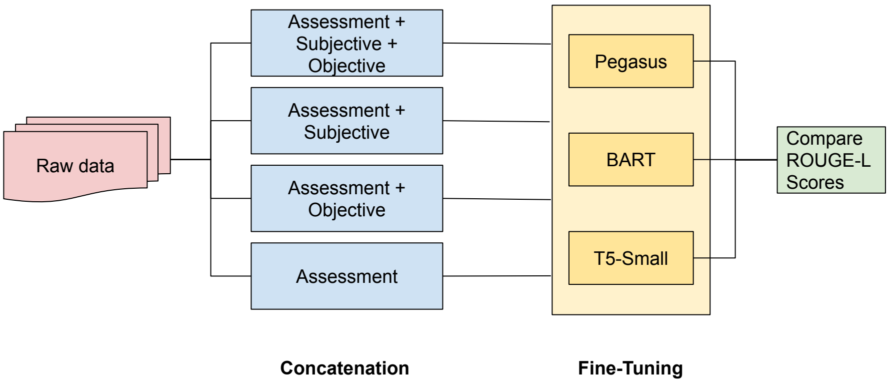

Figure 1: Feature Selection Architecture

NLP task (e.g., text summarization, machine translation, etc.). The model is composed of an encoder-decoder architecture. Two different versions of the model, namely "t5-small" and "t5-base", which are available on HuggingFace, were used in this task. The t5-small model has 60 million parameters, while the t5-base model has 220 million parameters. The t5-small model was finetuned for 20 epochs, while the t5-base model was finetuned for 10 epochs with a learning rate of 1e-4.

5 Experiments

1. **Abtractive vs Extractive summarization**: To check the feasibility of using extractive summarization, we calculated the number of common words between the input text and the summary in the training dataset. By dividing the number of common words by the total unique words in the summary, we get the percentage of words in the summary that are directly obtained from the input. We then take the mean of this metric across the entire training data. This gives us a percentage of 34%. From this value, we can deduce that even if we produce a near-perfect extractive summarization model, it will only have 34% words that are common with the words in the ground truths. Abstractive summarization does not have this barrier, since it can generate unseen words and in theory, produce summaries that match exactly with the ground truths.

2. **Examining the importance of different features**: We fine-tuned the abovementioned models on four data variations as shown in Figure 1. There are three input features in the dataset. We trained models on four combinations of these features and analyzed the scores to assess the relevance of the respective features. Features were combined by concatenating their text values and using custom tags as separators. We obtained the results of four models on each input variation and compared the average ROUGE score obtained on every input variation. The results for the same have been showcased in Table 1. We observe that models trained on only the assessment feature yield better results overall and hence we use only the assessment feature for further experimentation.

Feature	Average Rouge-L F1 Score
Assessment	0.2302
Assessment + Objective	0.2273
Assessment + Subjective	0.2184
Assessment + Objective + Subjective	0.2103

Table 1: Model scores on features

3. **Checking the effect of preprocessing**: We applied some basic pre-processing techniques to the data. We removed stop words and lemmatized the tokens. The objective behind these ablations was to remove noise occurring in the form of commonly occurring words and different forms of the same root word. We

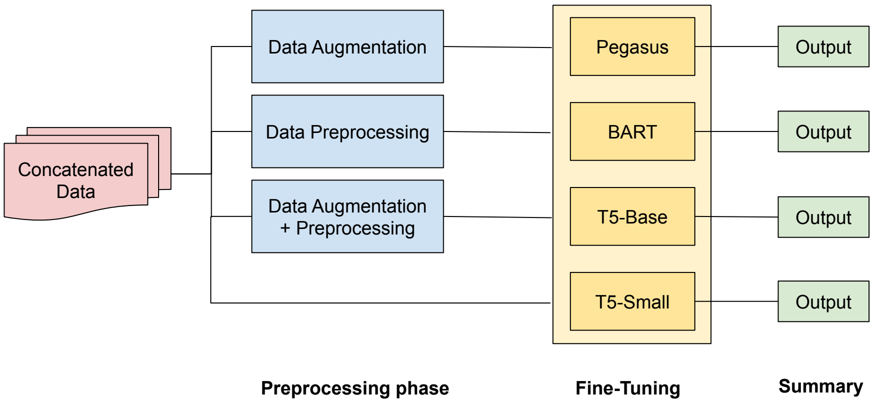

Figure 2: Model training on different input variations

observe a slight improvement in model performance due to these preprocessing techniques as shown in Table 2.

Input Variation	Model	Rouge-L F1 Score
Raw Dataset	BART	0.2512
	T5-small	0.1842
	T5-base	0.2370
	Pegasus	**0.2744**
Data Augmentation	BART	**0.2315**
	T5-small	0.1816
	T5-base	0.2158
	Pegasus	0.2307
Preprocessing	BART	**0.2519**
	T5-small	0.2134
	T5-base	0.2147
	Pegasus	0.2308
Data Augmentation + Preprocessing	BART	**0.2368**
	T5-small	0.1746
	T5-base	0.2261
	Pegasus	0.2152

Table 2: Model scores on input variations

4. **Augmenting the training data with MeQSUM dataset** We tried augmenting our training data with the MeQSUM (Abacha and Demner-Fushman, 2019) dataset. This dataset contains 1000 consumer health questions and their summaries. We create a new dataset that has 70% of our original data and 30% of the new MeQSUM data. We observe no significant change in model performance as shown in Table 2. We suspect this behavior is because although the domain of both datasets is similar, the structure of the inputs is vastly different. The dataset given for the task contains patient notes which are structurally different from the question-answer format in the MeQSUM dataset.

6 Conclusion

In this paper, we compared the performance of four models (Pegasus, BART, T5-base, and T5-small) on the task of summarizing patient daily care notes. Our experiments concluded that using only the assessment feature yielded better results on the ROUGE-L metric. We also observed an improvement in model performance by using preprocessing techniques like stop word removal and lemmatization. Overall, Pegasus outperformed other models yielding a score of 0.2744 on the test set. In the future, we plan to augment the data using data that is more similar to the task dataset. We also plan to implement an ensemble of the summarization models that we have used.

7 Limitations

The data augmentation method mentioned in the paper requires the augmentation dataset to be of a similar structure to the task dataset, which is not the case for MeQ-SUM. As a result, the data augmentation experiments do not provide significant improvement in performance.

References

Asma Ben Abacha and Dina Demner-Fushman. 2019. On the summarization of consumer health questions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2228–2234.

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 conference. Association for Computational Linguistics. North American Chapter. Meeting*, volume 2021, page 4794. NIH Public Access.

Stergos Afantenos, Vangelis Karkaletsis, and Panagiotis Stamatopoulos. 2005. Summarization from medical documents: a survey. *Artificial intelligence in medicine*, 33(2):157–177.

Muhammad Afzal, Fakhare Alam, Khalid Mahmood Malik, and Ghaus M Malik. 2020. Clinical context–aware biomedical text summarization using deep neural network: model development and validation. *Journal of medical Internet research*, 22(10):e19810.

Phyllis B Baxendale. 1958. Machine-made index for technical literature—an experiment. *IBM Journal of research and development*, 2(4):354–361.

Bharath Chintagunta, Namit Katariya, Xavier Amatriain, and Anitha Kannan. 2021. Medically aware gpt-3 as a data generator for medical dialogue summarization. In *Machine Learning for Healthcare Conference*, pages 354–372. PMLR.

Wesley T Chuang and Jihoon Yang. 2000. Extracting sentence segments for text summarization: a machine learning approach. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 152–159.

Harold P Edmundson. 1969. New methods in automatic extracting. *Journal of the ACM (JACM)*, 16(2):264–285.

Yanjun Gao, Dmitry Dligach, Timothy Miller, Matthew M. Churpek, 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 *Proceedings of the 22nd Workshop on Biomedical Language Processing*, Toronto, Canada. 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.

Virapat Kieuvongngam, Bowen Tan, and Yiming Niu. 2020. Automatic text summarization of covid-19 medical research articles using bert and gpt-2. *arXiv preprint arXiv:2006.01997*.

Ravikumar Kondadadi, Sahil Manchanda, Jason Ngo, and Ronan McCormack. 2021. Optum at mediqa 2021: Abstractive summarization of radiology reports using simple bart finetuning. In *Proceedings of the 20th Workshop on Biomedical Language Processing*, pages 280–284.

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*.

Julian Kupiec, Jan Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 68–73.

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*.

Hans Peter Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

Joel Larocca Neto, Alex A Freitas, and Celso AA Kaestner. 2002. Automatic text summarization using a machine learning approach. In *Advances in Artificial Intelligence: 16th Brazilian Symposium on Artificial Intelligence, SBIA 2002 Porto de Galinhas/Recife, Brazil, November 11–14, 2002 Proceedings 16*, pages 205–215. Springer.

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. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Zhongmin Shi, Gabor Melli, Yang Wang, Yudong Liu, Baohua Gu, Mehdi M Kashani, Anoop Sarkar, and Fred Popowich. 2007. Question answering summarization of multiple biomedical documents. In *Advances in Artificial Intelligence: 20th Conference of the Canadian Society for Computational Studies of Intelligence, Canadian AI 2007, Montreal, Canada, May 28-30, 2007. Proceedings 20*, pages 284–295. Springer.

Han-Chin Shing, Chaitanya Shivade, Nima Pourdamghani, Feng Nan, Philip Resnik, Douglas Oard, and Parminder Bhatia. 2021. Towards clinical encounter summarization: Learning to compose discharge summaries from prior notes. *arXiv preprint arXiv:2104.13498*.

Shengli Song, Haitao Huang, and Tongxiao Ruan. 2019. Abstractive text summarization using lstm-cnn based deep learning. *Multimedia Tools and Applications*, 78:857–875.

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.

Haoyu Zhang, Jianjun Xu, and Ji Wang. 2019. Pretraining-based natural language generation for text summarization. *arXiv preprint arXiv:1902.09243*.

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.